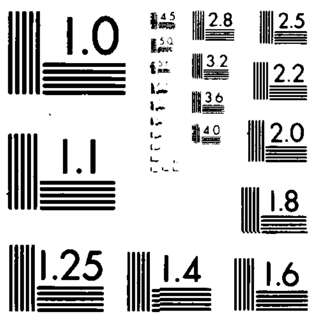
AD-A106 520   DOUGLAS AIRCRAFT CO . LONG BEACH CA   F/G 18/3
NUCLEAR BLAST RESPONSE COMPUTER PROGRAM. VOLUME III. PROGRAM LI--ETC(U)
AUG 81   J A MCGREW, H H CROXEN, T P KALMAN   DNA001-75-C-0216

UNCLASSIFIED   AFWL-TR-81-32-VOL-3   NL

1 of 5
AD
A106520



MICROCOPY RESOLUTION TEST CHART

NATIONAL BUREAU OF STANDARDS-1963-A

LEVEL III

AFWL-TR-81-32, Vol. III

2

AFWL-TR-
81-32
Vol. III

AD A106520

NUCLEAR BLAST RESPONSE COMPUTER PROGRAM

Volume III of III
Program Listing

J. A. McGrew, et al.

Douglas Aircraft Company
3855 Lakewood Blvd.
Long Beach, CA 90846

August 1981

Final Report

Approved for public release; distribution unlimited.

THIS RESEARCH WAS SPONSORED BY THE DEFENSE NUCLEAR AGENCY UNDER SUBTASK N99AXAE500, WORK UNIT 04, WORK UNIT TITLE: NUCLEAR BLAST RESPONSE COMPUTER PROGRAM.

AIR FORCE WEAPONS LABORATORY
Air Force Systems Command
Kirtland Air Force Base, NM 87117

This final report was prepared by the Douglas Aircraft Company, Long Beach, California under Contracts DNA 001-75-C-0216 and DNA 001-76-C-0346, Job Order 88090340 with the Air Force Weapons Laboratory, Kirtland Air Force Base, New Mexico. Mr. Alfred L. Sharp (NTYV) was the Laboratory Project Officer-in-Charge.

This report has been reviewed and is approved for publication.

Alfred L. Sharp

ALFRED L. SHARP
Project Officer

FOR THE COMMANDER

Kennedy B. Wilson

KENNEDY B. WILSON
Lt Colonel, USAF
Chief, Environment & Effects Branch

Norman K. Blocker

NORMAN K. BLOCKER
Colonel, USAF
Chief, Applied Physics Division

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AFWL-TR-81-32, Vol. III of III (VOL-3)
2. GOVT ACCESSION NO.: AD-A106 520
3. RECIPIENT'S CATALOG NUMBER:
4. TITLE (and Subtitle): NUCLEAR BLAST RESPONSE COMPUTER PROGRAM Volume III: Program Listing
5. TYPE OF REPORT & PERIOD COVERED: Final Report
6. PERFORMING ORG. REPORT NUMBER:
7. AUTHOR(s): J. A. McGrew, H. H. Croxen, T. P. Kalman, J. P. Giesing, W. P. Rodden
8. CONTRACT OR GRANT NUMBER(s): DNA 001-75-C-0216 and DNA 001-76-C-0346
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Douglas Aircraft Company, 3855 Lakewood Blvd., Long Beach, CA 90846
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 62601F/88090340
11. CONTROLLING OFFICE NAME AND ADDRESS: Air Force Weapons Laboratory (NTYV), Kirtland Air Force Base, NM 87117
12. REPORT DATE: August 1981
13. NUMBER OF PAGES: 434
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): Director, Defense Nuclear Agency, Washington, DC 20305
15. SECURITY CLASS. (of this report): Unclassified
15a. DECLASSIFICATION DOWNGRADING SCHEDULE:

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

This research was sponsored by the Defense Nuclear Agency under Subtask N99AXAE500, Work Unit 04, Work Unit Title: Nuclear Blast Response Computer Program.

(over)

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Aircraft Structural Response
Mass Modeling
Aerodynamic Modeling
Digital Computer Program
Gust Response
Blast
Nuclear Vulne
Dynamic Loads

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The VIBRA-6 computer program is a digital computer program developed to determine the response of aircraft to nuclear explosions when flying at subsonic speeds. It is similar to the VIBRA-4 program but uses the latest Doublet-Lattice Method for obtaining subsonic aerodynamic forces for arbitrary lifting surface-body configurations. The Doublet-Lattice procedure has been extended to account for the moving blast wave by considering it as a traveling gust. The nuclear blast representation remains the same as that used in the

18. SUPPLEMENTARY NOTES (Continued)

This report is divided into three volumes: Volume I contains the overall program descriptions and method of analysis, the input and output data descriptions, the program operation and a sample problem. Volume II details the unsteady aerodynamic procedure and Volume III contains the program listings.

20. ABSTRACT (Continued)

VIBRA-4 program but the method of solution of the equations of motion has been changed from that of numerical integration of quasi-steady equations of motion to a Fourier transform procedure to move from frequency domain solutions to time history solutions. The concept of dynamic core has been introduced to the program thus removing any restrictions on the size of the aircraft idealization which can be analyzed.

PREFACE

This report was prepared by the Douglas Aircraft Company, Long Beach, California, under Contract DNA 001-75-C-0216 and documents the subsonic unsteady aerodynamic module development for the revised VIBRA-6 Nuclear Blast Response Computer Program. This work was performed under Program Element NWE D 62704H, Project N99QAXA, Task Area B500, Work Unit 04 and was funded by the Defense Nuclear Agency under: RDT & E RMSS Code B342075464N99QAXAE50004H2590D. Funding of this effort was also supported by the Air Force Weapons Laboratory under: Program Element 62601F, Project 8809, Task 03, Work Unit 40.

Inclusive dates of research and development as documented herein were May 1975 through June 1977.

Volume I of this report documents the overall program descriptions and method of analysis, the input and output data descriptions, the program operation and a sample problem. Volume II of this report details the unsteady aerodynamic procedure.

J. A. McGrew was the program technical director for this task. The technical development was performed by J. P. Giesing and T. P. Kalman with the assistance of Dr. W. P. Rodden. The programming effort was carried out by T. P. Kalman and H. H. Croxen.

CONTENTS

Module	Subroutine	Page
CONTROL		1
	DACGUST	
	ATMOS	
	FDDTBL	
	HEADNG	
	MIS1	
	MIS2	
	PLT1	
	PRNT	
	RDTAPE	
	TRAERO	
	WRAERO	
	WRFRSP	
	WRLOAD	
	WRTAPE	
	WRUNIT	
AERODYNAMIC		45
	CSDLM	
	AERO	
	ATAN3	
	BFSMAT	
	DOTP	
	DPPS	
	DPZY	

CONTENTS (Cont'd)

Module	Subroutine	Page
	DUMULT	
	DYPZ	
	DZPY	
	DZYMAT	
	DZY	
	FLLD	
	FWMW	
	FZY2	
	GEND	
	GENF	
	GEOM	
	GRUP	
	GUST	
	HATS	
	IDF1	
	IDF2	
	INCRO	
	INPUTA	
	MATMUL	
	MUZYC	
	NACL	
	NEWH	
	ORGN	
	PISTON	

CONTENTS (Cont'd)

Module	Subroutine	Page
	READD	
	RHSIDE	
	ROWDYZ	
	RWREC	
	SB	
	SDLM	
	SNPDF	
	SOLVIT	
	SORT	
	SPLINE	
	SPLN	
	SPL1	
	SPL2	
	SUBB	
	SUBI	
	SUBP	
	TKER	
	TVOR	
	WANDWT	
	ZEROUT	
INERTIAL		226
	GENMS	
	INERTM	
	INPUTM	

CONTENTS (Cont'd)

Module	Subroutine	Page
	INPUTS	
	LHS	
	TMST	
	TMSTL	
UNIT LOADS		239
	LOAD	
	ABAMLT	
	AROLOD	
	AROPHA	
	BEAM	
	GEOMAB	
	GEOMMB	
	GEOMSB	
	INPUTL	
	MMBMLT	
	MSSPHI	
	RTOI	
	SBAMLT	
	SYMSOL	
	THRUST	
ACTIVE CONTROL		274
	ACS	
	ACSMOD	
	AFCS	

CONTENTS (Cont'd)

Module	Subroutine	Page
	TFBLIN	
	TFBLOK	
	TRFVAL	
FREQUENCY RESPONSE		286
	CFREQR	
	COEFF	
	FORMC	
	FREQRS	
	GSTINI	
	INTERP	
	TDTINT	
	TFMATF	
	TFMATM	
	TRFCMX	
UNIT GUST LOADS		309
	CFRLOD	
	FGMULT	
	FRLOAD	
	INTLPR	
TRIM		324
	AERO2	
	CTRIM	
	SPANLD	
	TRIM	

CONTENTS (Cont'd)

Module	Subroutine	Page
BLAST AND TIME RESPONSE		347
	ACCHST	
	CGUST	
	FLTPOS	
	GSTHST	
	GUSRHO	
	GUSTDR	
	IFT	
	INTRPB	
	LOADCH	
	PRESS	
	RANGE	
	STRSCH	
	TAR	
	TIMHST	
	TPEVAL	
	TPINT	
	TRFFT	
RIGID		396
	BLINT	
	CRIGID	
	GSINT	
	INTRP2	
	RIGID	

CONTENTS (Cont'd)

Module	Subroutine	Page
MERGE		412
	CMERGE	
	COPY	
	FCOPY	
	MERGE	

```
      PROGRAM DACGUST (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE31      DACGUST2
     2 , TAPE1=512,TAPE2,TAPE3=512,TAPE4=512,TAPE8=512,TAPE9=512         DACGUST3
     3 , TAPE10=512,TAPE11=512,TAPE12=512,TAPE13=512,TAPE14=512          DACGUST4
     4 , TAPE15=512,TAPE16=512,TAPE17=512,TAPE18=512,TAPE19=512          DACGUST5
     5 , TAPE20=512,TAPE21=512,TAPE22=512,TAPE23=512                     DACGUST6
     6 , TAPE34=512,TAPE35=512,TAPE36=512 )                              DACGUST7
C                                                                        DACGUST8
      COMMON NAA, A(1)                                                   DACGUST9
      COMMON/DISK2/ND2,ITBL2(843),NRECSA,IBUMP,NKD,VOBWS(20)             DISK2  2
      COMMON /ZZZ/HEDR(33),TITLE(12),DT(6),NIN,NOUT,KROW,LINES,IPRNT,NERDACGUS11
      COMMON /XTF/NF(100)                                                DACGUS12
      DIMENSION XF(100)                                                  DACGUS13
      EQUIVALENCE (NF(1),XF(1))                                          DACGUS14
      EQUIVALENCE (XF(5),NMS)                                            DACGUS15
     2     ,       (XF(6),NSYM),       (XF(7),NASYM),     (XF(8),NOMOD)  DACGUS16
     2     ,       (XF(9),NFREQ),      (XF(10),NENGS),    (XF(11),IDENT) DACGUS17
     3     ,       (XF(12),BZERO),     (XF(13),RHOO),     (XF(14),SIGMA) DACGUS18
     4     ,       (XF(15),EQUVAS),    (XF(16),IUNITS),   (XF(17),VKEAS) DACGUS19
     5     ,       (XF(18),VTFPS),     (XF(19),ALT)                      DACGUS20
     1     ,   (XF(24),KPRLDS),(XF(25),KPRTMH)                           DACGUS21
     2 , (XF(29),NACC)                                                   DACGUS22
     6     ,         (XF(70),MXORD),(XF(71),MXORSN),(XF(72),MXORSD)      DACGUS23
     7     , (NF(73),MXORAN),(NF(74),MXORAD),(NF(75),NTFS),(NF(76),NTFA) DACGUS24
     8 , (XF(22),KPRBLS),(XF(20),KPRCHK),(XF(21),KPRCXI)                 DACGUS25
     9 , (XF(100),SIZFCT)                                                DACGUS26
     1 ,   (XF(23),KPRTRM),(XF(51),AN    ),(XF(52),ZDOT   )              DACGUS27
     1 ,   (XF(53),RTURN ),(XF(54),KMAN   ),(XF(55),AB     )             DACGUS28
     1 ,   (XF(56),AC    ),(XF(57),INDSYM),(XF(61),NORMAX)               DACGUS29
     1 ,   (XF(62),TIMEMX),(XF(63),EFR    ),(XF(64),KGRD   )             DACGUS30
     1 ,   (XF(65),KLPT  ),(XF(66),HGRD ),(XF(67),KLOAD )                DACGUS31
     1 ,   (XF(68),NCRITS)                                               DACGUS32
     1 , (XF(31),NBEAMS),(XF(32),NINTLD),(XF(33),NSTRSS)                 DACGUS33
     1 , (XF(34),NMGRP ),(XF(35),NABGRP),(XF(36),NSBGRP)                 DACGUS34
      EQUIVALENCE (XF(26),IPLQ)                                          DACGUS35
     1     ,   (XF(27),IPLL)                                             DACGUS36
     1   ,   (XF(28),IPLBL)                                              DACGUS37
     1     ,       (XF(37),NBOXES),    (XF(38),NAERSB)                   DACGUS38
      EQUIVALENCE (XF(41),NK),         (XF(42),NG)                       DACGUS39
     1     ,       (XF(43),NBOX),      (XF(44),NSBETO),   (XF(45),NB)    DACGUS40
     1   ,   (XF(48),IDIMUL)                                             DACGUS41
     1 , (XF(1),AMACH)                                                   DACGUS42
     1 , (XF(3),VSS)                                                     DACGUS43
     1 , (XF(97),DELT)                                                   DACGUS44
     1 , (XF(99),RBRADF)                                                 DACGUS45
     1 , (XF(4),PO)                                                      DACGUS46
     1 , (XF(96),IPRNTM)                                                 DACGUS47
      EQUIVALENCE (XF(80),ISECT),    (XF(81),NDOF),     (XF(82),IMS)     DACGUS48
      COMMON /AEROMX/INTARO,NVBWMX,VOBWIN(400),RINTP(50,3)               DACGUS49
      COMMON /DDTBLS/DDTBL(20,10)                                        DACGUS50
      INTEGER DDTBL                                                      DACGUS51
      DIMENSION CR(3,20)                                                 DACGUS52
      DIMENSION UNITS(3)                                                 DACGUS53
```

```
      DIMENSION IHD(50),RHD(50)                                          DACGUS54
      DIMENSION OPDATI(6)                                                DACGUS55
      DIMENSION OPDATA(10), OPT(10)                                      DACGUS56
      EQUIVALENCE (IHD(1),RHD(1))                                        DACGUS57
      DATA AERO/4HAERO/, AMOD/4HIMOD/, ALOD/4HLOAD/, SECT/4HSECT/        DACGUS58
      DATA UNITS/1.0,0.59208578,0.68181818/                              DACGUS59
      DATA CORERQ/8H  MAIN  /                                            DACGUS60
      DATA CORERA/8HAERO STG/                                            DACGUS61
      DATA BLANK/4H    /                                                 DACGUS62
      DATA OPT/4HAERO,4HUNIT,4HACSM,4HFRSP,4HGUST, 4HBLST,4HRIGD         DACGUS63
     1 , 4HMERG, 4H    , 4H    /, NOPT/8/                                DACGUS64
C                                                                        DACGUS65
      NIN = 5                                                            DACGUS66
      NOUT = 6                                                           DACGUS67
      KROW = 54                                                          DACGUS68
      IPRNT = 1                                                          DACGUS69
      LINES = 0                                                          DACGUS70
C                                                                        DACGUS71
      NTI = 31                                                           DACGUS72
      NP2 = 2                                                            DACGUS73
C                                                                        DACGUS74
      DDTBL(1,1) = 19                                                    DACGUS75
      DDTBL(2,1) = 4HAERO                                                DACGUS76
      DDTBL(3,1) = 2                                                     DACGUS77
      DDTBL(1,5) = 35                                                    DACGUS78
      DDTBL(2,5) = 4HFRSP                                                DACGUS79
      DDTBL(3,5) = 2                                                     DACGUS80
      DDTBL(1,4) = 34                                                    DACGUS81
      DDTBL(2,4) = 4HUNIT                                                DACGUS82
      DDTBL(3,4) = 2                                                     DACGUS83
      DDTBL(1,6) = 36                                                    DACGUS84
      DDTBL(2,6) = 4HLOAD                                                DACGUS85
      DDTBL(3,6) = 2                                                     DACGUS86
C                                                                        DACGUS87
      DO 10 I=1,100                                                      DACGUS88
   10 NF(I) = 0                                                          DACGUS89
C                                                                        DACGUS90
      NAA = 1                                                            DACGUS91
      IUNITS=2                                                           DACGUS92
C                                                                        DACGUS93
C     HERE STARTS THE RUN DATA DECK INPUT                                DACGUS94
C                                                                        DACGUS95
      READ(NIN,300)IDENT,KPRCXQ,KPRCHQ,KPRCXL,KPRCHL                     DACGUS96
C                                                                        DACGUS97
      IF (IDENT.LE.0) GO TO 290                                          DACGUS98
      CALL HEADNG                                                        DACGUS99
      WRITE(NOUT,320)IDENT                                               DACGU100
C                                                                        DACGU101
      READ(NIN,305)(OPDATI(I),I=1,6)                                     DACGU102
C                                                                        DACGU103
  305 FORMAT(6(A4,4X))                                                   DACGU104
C                                                                        DACGU105
```

```
      DO 75 I=1,NOPT
   75 OPDATA(I)=BLANK
      DO 80 I=1,6
      DO 80 J=1,NOPT
      IF (OPDATI(I).EQ.OPT(J)) OPDATA(J) = OPT(J)
   80 CONTINUE
C
      WRITE (NOUT,600) (OPDATA(I),I=1,NOPT)
  600 FORMAT(1H0,20X, THE FOLLOWING ANALYSIS CODES HAVE BEEN ,
     1 *CALLED FOR IN THIS RUN*/1H0,20X,10(A4,4X))
C
      IPRNTM=0
      DO 85 I=2,4
   85 IF(OPDATA(I).NE.BLANK)IPRNTM=1
C
      IF (OPDATA(8).NE.BLANK) GO TO 280
      IF(OPDATA(1).EQ.BLANK)GO TO 105
C
   90 FORMAT (A4)
C
      CALL CSDLM (NTI)
C
      DO 101 I=2,7
      IF (OPDATA(I).NE.BLANK) GO TO 105
  101 CONTINUE
      GO TO 290
C
  105 CALL RDAERO (4HHEAD,0,IHD,NOUT,NER)
C     NER EQUALS 0 MEANS NO END OF FILE AND AERO ALREADY EXISTS
      IF (NER.EQ.0) GO TO 110
C
      WRITE (NOUT,602)
  602 FORMAT (1H1,20X, AERO FILE IS NOT DEFINED  -  STOP )
      GO TO 290
C
  110 NK     = IHD(2)
      NKO    = IHD(2)
      NSYM   = IHD(3)
      NASYM  = IHD(4)
      NG     = IHD(5)
      NB     = IHD(6)
      NBOX   = IHD(7)
      NSBETO = IHD(8)
      BZERO = RHD(24)/2.0
C
      READ(NIN,310)ALT,VKEAS,SIZFCT,AIPLQ,AIPLL
      IPLQ=AIPLQ
      IPLL=AIPLL
      IF(SIZFCT.EQ.0)SIZFCT=1.0
C
C     SIZFCT IS 1.0 IF ALL GEOM DATA INPUT IN INCHES
C     IF NOT, THEN SIZFCT=SIZE IN INCHES/SIZE IN YOUR UNITS
```

```
C      BUT REMEMBER MASS IS MASS
C
       IF(VKEAS.LE.0)STOP
C
       CALL ATMOS(ALT,TEMP,SIGMA,RHO,TR,PO,VSS,AMU,KK)
       RHOO=RHO/SIGMA
       EQUVAS=SQRT(SIGMA)
       VTFPS=VKEAS/(UNITS(IUNITS)*EQUVAS)
       DYNP=0.50*RHO*VTFPS**2
       PO=14.696*PO
       AMACH=VTFPS/VSS
       CALL HEADNG
       WRITE(NOUT,360)VKEAS,ALT,SIZFCT,VTFPS,DYNP,SIGMA,AMACH
     1 , IPLQ,IPLL
C
       IF(AMACH.LT.1.0)GO TO 200
       WRITE(NOUT,770)
  770 FORMAT(1H1,20X, OH OH MACH IS OVER 1.0     STOP )
       STOP
C
  200 CONTINUE
C
       IF (OPDATA(7).NE.BLANK) GO TO 270
C
C      FIND THE IMOD OR SECT INPUT DATA
C
  120 READ (NTI,90) CHECK
       IF (CHECK.NE.AMOD) GO TO 122
       ISECT = 0
       IMS = 1
       NDOF = 3
C
C      IMOD INPUT DATA FOUND
C
       READ (NTI,300) NMS, NENGS, KPRLDS
       WRITE(NOUT,370)NMS,NSYM,NASYM,NENGS,KPRLDS
  370 FORMAT(1H0,20X, INERTIAL DATA /1H0,
     1 20X, NO. TOTAL MASSES           = ,I4/ 1H ,
     1 20X, NO. SYMMETRIC MODES        = ,I4/ 1H ,
     1 20X, NO. ANTISYMMETRIC MODES    = ,I4/ 1H ,
     1 20X, NO. ENGINES                = ,I4/ 1H ,
     1 20X, UNIT LOAD PRNT FLAG        = ,I4)
       GO TO 125
C
  122 IF (CHECK.NE.SECT) GO TO 120
       ISECT = 1
       READ (NTI,300) NMS, NDOF, MSYM, MASYM, NENGS, KPRLDS
C
C *** CHECK MSYM AND MASYM
C
       IF (NSYM.EQ.MSYM.AND.NASYM.EQ.MASYM) GO TO 124
       NFP = 1
```

```
      WRITE (NOUT,123)                                                  DACGU210
  123 FORMAT (1H0,*NSYM AND/OR NASYM INCOMPATIBLE WITH THE AERO FILE*)  DACGU211
      GO TO 290                                                         DACGU212
C                                                                       DACGU213
  124 IMS = 10                                                          DACGU214
      IF (NDOF.EQ.7) IMS = 16                                           DACGU215
      IF (NDOF.EQ.8) IMS = 23                                           DACGU216
  125 CONTINUE                                                          DACGU217
C                                                                       DACGU218
C     NOW READ IN ACS DATA IF RQD                                       DACGU219
C                                                                       DACGU220
      NTFS=0                                                            DACGU221
      NTFA=0                                                            DACGU222
      MXBLK=0                                                           DACGU223
      MXOBLK=0                                                          DACGU224
      MXORD=0                                                           DACGU225
C                                                                       DACGU226
      IF(OPDATA(3).EQ.BLANK)GO TO 205                                   DACGU227
C                                                                       DACGU228
      READ(NIN,300)NTFS,NTFA,MXBLK,MXOBLK                               DACGU229
C                                                                       DACGU230
      WRITE(NOUT,511)NTFS,NTFA,MXBLK,MXOBLK                             DACGU231
C                                                                       DACGU232
      MXOBLK=MXOBLK+1                                                   DACGU233
      MXORD=MXOBLK*MXBLK                                                DACGU234
C                                                                       DACGU235
  205 CONTINUE                                                          DACGU236
C                                                                       DACGU237
      NOMOD = NSYM+NASYM                                                DACGU238
      IDIMUL=1                                                          DACGU239
      IF(OPDATA(2).EQ.BLANK)IDIMUL=0                                    DACGU240
C                                                                       DACGU241
      L1 = 1                                                            DACGU242
      L2  = L1  + 20                                                    DACGU243
      L3  = L2  + NOMOD                                                 DACGU244
      L4  = L3  + NOMOD*NOMOD                                           DACGU245
      L41 = L4  + NOMOD                                                 DACGU246
      LMINB = L41                                                       DACGU247
      L9 = L41 + NOMOD                                                  DACGU248
      L10 = L9 + NMS*NOMOD                                              DACGU249
      L11 = L10 + NMS*NOMOD                                             DACGU250
      LMODE = L11 + NMS*NOMOD                                           DACGU251
      IF (NDOF.NE.3) LMODE = L9 + NDOF*NMS*NOMOD                        DACGU252
      L51 = LMODE                                                       DACGU253
      L52=L51+NTFS*4                                                    DACGU254
      L53=L52+NTFA*4                                                    DACGU255
      L54=L53+2*NTFS*MXORD                                              DACGU256
      L55=L54+2*NTFA*MXORD                                              DACGU257
      L56=L55+NTFS*NSYM                                                 DACGU258
      LFRQRS = L56 + NTFA*NASYM                                         DACGU259
      L5 = LFRQRS                                                       DACGU260
      L6  = L5  + IMS*NMS                                               DACGU261
```

```
      L7  = L6  + NMS
      L8  = L7  + NMS
      L12 = L8 + NMS
C
      L13 = L12 + NDOF*NMS*NOMOD*IDIMUL
      L14 = L13 + 2*NENGS*IDIMUL
      L15 = L14 + NDOF*NDOF*NMS*ISECT
      NAA = L15
      WRITE(NOUT,999)NAA,CORERQ
C
      CALL INERTM(NTI,NMS,
     1 A(L5),A(L6),A(L7),A(L8),A(L2),A(L9),A(L10),A(L11),
     2 A(L14),
     1 A(L1),A(L41),A(L12),A(L3),A(L4),NOMOD,NSYM,NASYM)
C
      CALL HEADNG
      IF(OPDATA(2).EQ.BLANK)GO TO 207
C
C     FIND THE LOAD INPUT DATA
C
  130 READ (NTI,90) CHECK
      IF (CHECK.NE.ALOD) GO TO 130
C     LOAD INPUT DATA FOUND
C
C
C     READ IN HERE THE CONTROLLING DATA FOR UNIT LOADS
C
      READ (NTI,300) NBEAMS,NINTLD,NSTRSS,NMGRP,NABGRP,NSBGRP
C
      WRITE(NOUT,345)NBEAMS,NINTLD,NSTRSS,NMGRP,NABGRP,NSBGRP
C
  345 FORMAT(1H0,20X, UNIT LOAD DATA /1H0,
     1 20X, NO. INTGD LOAD BEAMS       =  ,I4/1H ,
     1 20X, NO. INTGD LOADS DEFINED    =  ,I4/1H ,
     1 20X, NO. STRESSES               =  ,I4/1H ,
     1 20X, NO. MASS GROUPS            =  ,I4/1H ,
     1 20X, NO. AERO BOX GROUPS        =  ,I4/1H ,
     1 20X, NO. SLENDER BODY GROUPS    =  ,I4)
C
      CALL LOAD (NTI, A(L6), A(L7), A(L8), A(L12), NDOF,NMS,NSYM,NASYM,
     1 A(L9),A(L10),A(L11),NENGS,A(L13))
      IF (NER.NE.0) GO TO 290
C
C
      GO TO 220
C
  207 IF (OPDATA(5).EQ.BLANK.AND.OPDATA(6).EQ.BLANK) GO TO 220
C
      CALL RDUNIT (1,0,IHD,NOUT,NER)
C     NER EQUALS 0 MEANS NO END OF FILE AND UNIT ALREADY EXISTS
      IF (NER.EQ.0) GO TO 210
C
```

```
      WRITE  (NOUT,603)                                                 DACGU314
  603 FORMAT (1H1,20X, UNIT FILE IS NOT DEFINED  -   STOP )             DACGU315
      GO TO 290                                                         DACGU316
C                                                                       DACGU317
  210 NINTLD = IHD(3)                                                   DACGU318
      NSTRSS = IHD(4)                                                   DACGU319
      NENGS  = IHD(5)                                                   DACGU320
      NAPGRP = IHD(9)                                                   DACGU321
      NSHGRP = IHD(10)                                                  DACGU322
      NBOXES = IHD(11)                                                  DACGU323
      NAERSB = IHD(12)                                                  DACGU324
C                                                                       DACGU325
  220 CONTINUE                                                          DACGU326
C                                                                       DACGU327
      IF(OPDATA(3).EQ.BLANK)GO TO 225                                   DACGU328
C                                                                       DACGU329
      IF(NTFS.NE.0.OR.NTFA.NE.0)CALL  ACS(                              DACGU330
     1 NIN,MXBLK,MXOBLK,A(L9),A(L10),A(L11),                            DACGU331
     1 A(L51),A(L52),A(L53),A(L54),A(L55),A(L56))                       DACGU332
C                                                                       DACGU333
  225 CONTINUE                                                          DACGU334
C                                                                       DACGU335
      IF  (OPDATA(4).EQ.BLANK.AND.OPDATA(5).EQ.BLANK.AND.               DACGU336
     1     OPDATA(6).EQ.BLANK) GO TO 290                                DACGU337
C                                                                       DACGU338
      NAA  =  LERORS                                                    DACGU339
C                                                                       DACGU340
      NTOTAP=NBOX+2*NSBETO                                              DACGU341
      MTOTAP=2*NTOTAP                                                   DACGU342
C                                                                       DACGU343
      L60  =  NAA                                                       DACGU344
      L61  =  L60                                                       DACGU345
      LENGTH=MAX0(2*NSYM*NSYM,2*NASYM*NASYM,NTOTAP*(NSYM+NASYM),        DACGU346
     1 2*NTOTAP*NG)                                                     DACGU347
      L62  =  L61  +  LENGTH                                            DACGU348
      L63  =  L62                                                       DACGU349
      L64  =  L63  +  NTOTAP*NSYM                                       DACGU350
      LTAERO  =  L64  +  NTOTAP*NASYM                                   DACGU351
C                                                                       DACGU352
      WRITE(NOUT,999)LTAERO,CORERA                                      DACGU353
C                                                                       DACGU354
C                                                                       DACGU355
      CALL  TRAERO  (NOUT,NER,MTOTAP,NTOTAP                             DACGU356
     1  , CR,A(L60),A(L61),A(L62),A(L63),A(L64)  )                      DACGU357
      IF  (NER.NE.0)  GO TO 290                                         DACGU358
C                                                                       DACGU359
      IF(OPDATA(4).EQ.BLANK)GO TO 227                                   DACGU360
C                                                                       DACGU361
C     NOW READ IN FREQS FOR FREQ RESPNSE SOLN IF RQD                    DACGU362
C                                                                       DACGU363
      READ(NIN,300)NFRGR                                                DACGU364
      I  =  0                                                           DACGU365
```

```
      DO 150 N=1,NFRGR                                                   DACGU366
      READ (NIN,310) (RINTP(N,J),J=1,3)                                  DACGU367
      I = I+1                                                            DACGU368
      VOBWIN(I) = RINTP(N,1)                                             DACGU369
  140 I = I+1                                                            DACGU370
      VOBWIN(I) = VOBWIN(I-1) + RINTP(N,3)                               DACGU371
      IF (VOBWIN(I).LT.RINTP(N,2)) GO TO 140                             DACGU372
      IF(VOBWIN(I).GT.RINTP(N,2))I=I-1                                   DACGU373
  150 NFREQ = I                                                          DACGU374
C                                                                        DACGU375
      WRITE(NOUT,30)NFREQ                                                DACGU376
      DO 155 I=1,NFRGR                                                   DACGU377
  155 WRITE(NOUT,31)I,(RINTP(I,J),J=1,3)                                 DACGU378
C                                                                        DACGU379
   30 FORMAT(1H0,20X, SOLUTION FREQUENCY REQUEST FOR ,I4,  FREQUENCIES /DACGU380
     1 1H0, ITM ,6X, LWR F ,6X, UPR F ,6X, DEL F )                       DACGU381
   31 FORMAT(1H ,I3,3(1X,F10.4))                                         DACGU382
C                                                                        DACGU383
      KPRCXI=KPRCXQ                                                      DACGU384
      KPRCHK=KPRCHQ                                                      DACGU385
C                                                                        DACGU386
      CALL CFREQR (CR,A(L1),A(L2),A(L3),A(L4),A(L41),N3MOD,NFREQ,NG,     DACGU387
     1    A(L51),A(L52),A(L53),A(L54),A(L55),A(L56)                      DACGU388
     2  )                                                                DACGU389
C                                                                        DACGU390
  227 IF (OPDATA(5).EQ.BLANK) GO TO 247                                  DACGU391
C                                                                        DACGU392
      CALL RDFRSP (1,0,0,IHD,NOUT,NER)                                   DACGU393
C     NER EQUALS 0 MEANS NO END OF FILE AND FRSP ALREADY EXISTS          DACGU394
      IF (NER.EQ.0) GO TO 230                                            DACGU395
C                                                                        DACGU396
      WRITE (NOUT,604)                                                   DACGU397
  604 FORMAT (1H1,20X, FRSP FILE IS NOT DEFINED  -  STOP )               DACGU398
      GO TO 290                                                          DACGU399
C                                                                        DACGU400
  230 NFREQ = IHD(4)                                                     DACGU401
      NK = IHD(7)                                                        DACGU402
      VTFPS = RHD(11)                                                    DACGU403
      SIGMA = RHD(12)                                                    DACGU404
C                                                                        DACGU405
      READ (NIN,300) NACC                                                DACGU406
C                                                                        DACGU407
      NAA=LFRQRS                                                         DACGU408
C                                                                        DACGU409
      KPRCXI=KPRCXL                                                      DACGU410
      KPRCHK=KPRCHL                                                      DACGU411
C                                                                        DACGU412
      CALL CFRLOD(CR,                                                    DACGU413
     1 A(L9),A(L10),A(L11),NMS,                                          DACGU414
     1 A(L51),A(L52),A(L53),A(L54),A(L55),A(L56))                        DACGU415
C                                                                        DACGU416
  247 IF (OPDATA(6).EQ.BLANK) GO TO 290                                  DACGU417
```

```
C                                                                       DACGU418
      CALL RDLOAD (1,0,IHD,NOUT,NER)                                    DACGU419
C     NER EQUALS 0 MEANS NO END OF FILE AND LOAD ALREADY EXISTS         DACGU420
      IF (NER.EQ.0) GO TO 250                                           DACGU421
C                                                                       DACGU422
      WRITE (NOUT,605)                                                  DACGU423
  605 FORMAT (1H1,20X, LOAD FILE IS NOT DEFINED  -  STOP )              DACGU424
      GO TO 290                                                         DACGU425
C                                                                       DACGU426
  250 NFREQ = IHD(4)                                                    DACGU427
      NINTLD = IHD(5)                                                   DACGU428
      NACC = IHD(6)                                                     DACGU429
C                                                                       DACGU430
      KPRCHK=0                                                          DACGU431
      KPRCXI=0                                                          DACGU432
      NAA = LMINB                                                       DACGU433
C     IF RB ROT MODES ARE FOR AN ANGLE OTHER THAN 1 RADIAN INPUT        DACGU434
C     RATIO OF 1 RADIAN/RADIANS USED FOR TRIM SOLN CORRECTION           DACGU435
C                                                                       DACGU436
C                                                                       DACGU437
      READ(NIN,310)AN,AKMAN,ZDOT,AKPRCH,RBRADF                          DACGU438
      KMAN=AKMAN                                                        DACGU439
      KPRTRM=AKPRCH                                                     DACGU440
C                                                                       DACGU441
      READ(NIN,310)TIMEMX,DELT,EFR,HGRD,AKPRTM,AIPLBL                   DACGU442
      IPLBL=AIPLBL                                                      DACGU443
      KPRTMH=AKPRTM                                                     DACGU444
C                                                                       DACGU445
      READ(NIN,300)NORMAX,KGRD,KLPT,KLOAD,NCRITS,KPRBLS                 DACGU446
C                                                                       DACGU447
      CALL HEADNG                                                       DACGU448
      WRITE(NOUT,60)AN,KMAN,ZDOT,KPRTRM,RBRADF                          DACGU449
C                                                                       DACGU450
      WRITE(NOUT,70)TIMEMX,EFR,HGRD,KPRTMH,NORMAX,KGRD,KLPT,KLOAD,      DACGU451
     1 NCRITS,KPRBLS,IPLBL,DELT                                         DACGU452
C                                                                       DACGU453
      CALL CGUST (CR,A(L1),A(L3),A(L4) )                                DACGU454
C                                                                       DACGU455
      GO TO 290                                                         DACGU456
C                                                                       DACGU457
  270 CALL CRIGID                                                       DACGU458
      GO TO 290                                                         DACGU459
C                                                                       DACGU460
  280 CALL CMERGE                                                       DACGU461
C                                                                       DACGU462
  290 STOP                                                              DACGU463
C                                                                       DACGU464
C                                                                       DACGU465
300   FORMAT (6I12)                                                     DACGU466
310   FORMAT (6F12.1)                                                   DACGU467
  320 FORMAT(1H0,20X, FIXED DATA DECK INPUT GO /1H0,20X, IDENT= ,I12)   DACGU468
  360 FORMAT(1H0,20X, RUN DATA DECK INPUT GO /1H0,                      DACGU469
```

```
    1 20X, FLIGHT DATA FOR FREQUENCY RESPONSE AND UNIT GUST LOADS /1H0,DACGU470
    1 20X, VKEAS          =   ,F8.3,   KEAS /1H ,                        DACGU471
    1 20X, ALTITUDE       =   ,F10.3,   FT. /1H ,                        DACGU472
    1 20X, SIZE FACTOR =   ,F8.4,   INCHES/UNIT /1H ,                    DACGU473
    1 20X, VTRUE          =   ,F8.3,   FPS /1H ,                         DACGU474
    1 20X, DYNAMIC P      =   ,F8.3,   PSF /1H ,                         DACGU475
    1 20X, SIGMA          =   ,E12.5/1H ,                                DACGU476
    1 20X, MACH           =   ,F8.4/1H ,                                 DACGU477
    1 20X, PLOT Q FLG  =   ,I4/1H ,                                      DACGU478
    1 20X, PLOT GST FLG=   ,I4)                                          DACGU479
511 FORMAT(1H0,20X, ACTIVE CONTROL SYSTEM DATA /1H0,                     DACGU480
    1 20X, NTFS       = ,I4/1H ,                                         DACGU481
    1 20X, NTFA       = ,I4/1H ,                                         DACGU482
    1 20X, MXBLK      = ,I4/1H ,                                         DACGU483
    1 20X, MXOBLK     = ,I4)                                             DACGU484
999 FORMAT(1H0,20X,I10,  WORDS OF CORE RQD FOR STEP +++ ,A10, +++ )      DACGU485
60  FORMAT(1H0,20X, RUN DATA FOR TRIM /1H0,                              DACGU486
    1 20X, LOAD FACTOR         = ,F8.3/1H ,                              DACGU487
    1 20X, MANEUVER CODE       = ,I4  /1H ,                              DACGU488
    1 20X, ZDOT                = ,F8.3,   FPS /1H ,                      DACGU489
    1 20X, TRIM MTX PRNT FLG  = ,I4/1H ,                                 DACGU490
    1 20X, RB MODE ROT. CORR. = ,E12.5)                                  DACGU491
70  FORMAT(1H0,20X, RUN DATA FOR BLAST /1H0,                             DACGU492
    1 20X, MAX SOLN TIME           = ,F8.3,  SECS /1H ,                  DACGU493
    1 20X, YIELD                   = ,F8.3,  KT   /1H ,                  DACGU494
    1 20X, GROUND HEIGHT           = ,F8.1,  FT.  /1H ,                  DACGU495
    1 20X, LOAD TIME HSTRY PRNT = ,I4  /1H ,                             DACGU496
    1 20X, NO. ORIENTATIONS        = ,I4  /1H ,                          DACGU497
    1 20X, GROUND REFL. CODE       = ,I4  /1H ,                          DACGU498
    1 20X, ITERATION CODE          = ,I4  /1H ,                          DACGU499
    1 20X, MOD CRIT LOAD CODE      = ,I4  /1H ,                          DACGU500
    1 20X, MOD CRIT STRSS CODE     = ,I4  /1H ,                          DACGU501
    1 20X, LOAD MTX PRNT FLG       = ,I4/1H ,                            DACGU502
    1 20X, PLOT BLST LOAD FLG      = ,I4/1H ,                            DACGU503
    1 20X, INITIAL DELTA T         = ,F10.5,  SECS )                     DACGU504
      END                                                                DACGU505
```

```
      SUBROUTINE ATMOS(Z,TM,SIGMA,RHO,THETA,DELTA,CA,AMU,K)
C     CALLING SEQUENCE
C     CALL ATMOS(Z,TM,SIGMA,RHO,THETA,DELTA,CA,AMU,K)
C     Z     = GEOMETRIC ALTITUDE (FT)
C     TM    = MOLECULAR SCALE TEMPERATURE (DEGREES RANKINE)
C     THETA = RATIO OF TEMPERATURE TO THAT AT SEA LEVEL
C     DELTA = RATIO OF PRESSURE TO THAT AT SEA LEVEL
C     CA    = SPEED OF SOUND (FT/SEC)
C     AMU   = VISCOSITY COEFFICIENT (LB-SEC/FT**2)
C     K = 1 NORMAL,
C       = 2 ALTITUDE GREATER THAN 300000. FT.,
C       = 3 ALTITUDE NEGATIVE,
C       = 4 FLOATING POINT OVERFLOW,
C       = 5 ALTITUDE GREATER THAN 300000. FT. AND FLOATING POINT OVERFL.
      DIMENSION HPRIMB(11), TMB(11), SIGMAB(11), ALM(11)
      DATA HPRIMB/0.0,36089.239,82020.997,154199.480,173884.510,259186.3
     150,295275.590,344488.190,524934.380,557742.780,656167.980/,TMB/518
     2.688,389.988,389.988,508.788,508.788,298.188,298.188,406.188,2386.
     3188,2566.188,2836.188/,SIGMAB/1.00,2.9706958E-01,3.2665751E-02,1.2
     4117870E-03,5.8677311E-04,1.7329156E-05,1.7928595E-06,9.3921519E-08
     5,7.7658593E-10,5.6324877E-10,2.5726771E-10/,ALM/-0.00356616,0.0,0.
     6001645920,0.0,-0.00246888,0.0,0.00219456,0.01097280,0.00548640,0.00
     7274320,0.00192024/
      DATA Q/0.018744176/,RE/2.0855531E07/,S/198.72/,AMUZ/3.7372998E-07/
     1,RHOZ/0.0023769/,TMZ/518.688/
      K=1
      IF (Z) 10,30,20
   10 K=3
      GO TO 110
   20 IF (Z.GT.300000.) K=K+1
   30 HPRIM=(RE/(RE+Z))*Z
      DO 40 M=1,11
      IF (HPRIM-HPRIMB(M)) 50,60,40
   40 CONTINUE
      M=12
   50 M=M-1
   60 IF (ALM(M)) 70,80,70
   70 TM=TMB(M)+ALM(M)*(HPRIM-HPRIMB(M))
      SIGMA=EXP((1.0+(Q/ALM(M)))*(ALOG(TMB(M)/TM)))*SIGMAB(M)
      GO TO 90
   80 TM=TMB(M)
      SIGMA=SIGMAB(M)*EXP(-(Q*(HPRIM-HPRIMB(M)))/TMB(M))
   90 RHO=RHOZ*SIGMA
      THETA=TM/TMZ
      DELTA=SIGMA*THETA
      CA=49.02177*SQRT(TM)
      AMU=AMUZ*SQRT(THETA**3)*((TMZ+S)/(TM+S))
  100 K=K+3
  110 RETURN
      END
```

```
      SUBROUTINE FDDTBL (CH,JI,JO)                                      FDDTBL 2
      COMMON /DDTBLS/DDTBL(20,10)                                       FDDTBL 3
      INTEGER DDTBL, CH                                                 FDDTBL 4
      JI = 0                                                            FDDTBL 5
      JO = 0                                                            FDDTBL 6
      DO 100 J=1,10                                                     FDDTBL 7
      IF (DDTBL(2,J).NE.CH) GO TO 100                                   FDDTBL 8
      IF (DDTBL(3,J).NE.3) JI=J                                         FDDTBL 9
      IF (DDTBL(3,J).NE.1) JO=J                                         FDDTBL10
  100 CONTINUE                                                          FDDTBL11
      RETURN                                                            FDDTBL12
      END                                                               FDDTBL13
```

```
      SUBROUTINE HEADNG                                                 HEADNG 2
      COMMON /ZZZ/CASE(30),X(12),DATE(2),TIME(2),Y,Z,NIN,NOUT,W,LINES   HEADNG 3
    1 FORMAT (1H1,T88,4HDATE,2X2A4,T2,15A4/1X15A4,T88,4HTIME,2X2A4)     HEADNG 4
      WRITE (NOUT,1) DATE,CASE,TIME                                     HEADNG 5
      LINES = 2                                                         HEADNG 6
      RETURN                                                            HEADNG 7
      END                                                               HEADNG 8
```

```
      SUBROUTINE MISI ( A, N, ND, B, MD, NERR, D )
C
C     MISI - MATRIX INVERT OR SOLVE
C
C     THIS SUBROUTINE SOLVES THE MATRIX EQUATION  A*C = B  WHERE A AND
C          B ARE INPUT.  IF B IS A UNIT MATRIX, THEN C IS THE INVERSE
C          OF A.
C
C          A      = IN-OUT = A TWO DIMENSIONAL ARRAY (DIMENSIONED ND BY
C                            ND) CONTAINING AN INPUT MATRIX OF ORDER N.
C                            THE A MATRIX IS DESTROYED BY THE SUBROUTINE.
C
C          N      = INPUT  = THE ORDER (INTEGER) OF A (N LESS THAN OR
C                            EQUAL TO ND) AND THE NUMBER OF ROWS IN B.
C
C          ND     = INPUT  = THE MAXIMUM DIMENSIONS ( INTEGER ) OF THE
C                            SQUARE MATRIX A.
C
C          B      = IN-OUT = A TWO DIMENSIONAL ARRAY (DIMENSIONED ND BY
C                            MD) CONTAINING THE SECOND INPUT MATRIX.
C                            THIS ARRAY IS MODIFIED AND CONTAINS (UPON
C                            OUTPUT) THE C MATRIX SOLUTION.
C
C          MD     = INPUT  = THE NUMBER (INTEGER) OF COLUMNS IN B.
C
C          NERR = OUTPUT = OUTPUT CODE (INTEGER)
C                        = 0   IF A IS NON-SINGULAR
C                        = 1   IF A IS SINGULAR
C                        = 2   IF N IS LESS THAN OR EQUAL TO ZERO
C                        = 3   IF N IS GREATER THAN ND
C
C          D      = INPUT  = SCALE FACTOR TO SCALE DETERMINENT.
C                 = OUTPUT = THE SCALED DETERMINENT OF A.
C
      DIMENSION  A(ND,ND), B(ND,MD)
      EQUIVALENCE   (L,FL), (K,FK)
C
C     CHECK THE VALUE OF N
C
      IF   ( N - 1 )10,20,30
C
   10 D = 0.0E0
      NERR = 2
      GO TO 190
C
   20 AIJMAX = A(1,1)
      IF   ( AIJMAX .EQ. 0.0E0 )   GO TO 60
      B(1,1) = B(1,1) / AIJMAX
      D = D * AIJMAX
      GO TO 180
C
   30 IF   ( N .LE. ND )   GO TO 40
```

```
      NERR = 2
      GO TO 190
C
C     START REDUCTION OF MATRIX A
C
   40 DO  130  I = 1, N
C
C     SEARCH FOR MAXIMUM ELEMENT IN ITH ROW OF A-MATRIX
C
      AIJMAX = A(I,1)
      JMAX = 1
      DO  50   J = 2, N
      IF   ( ABS( A(I,J) ) .LE. ABS( AIJMAX ) )   GO TO 50
      AIJMAX = A(I,J)
      JMAX = J
   50 CONTINUE
C
C     IF AIJMAX IS ZERO, THE MATRIX IS SINGULAR
C
      IF   ( ABS( AIJMAX ) .GT. 0.0E0 )   GO TO 70
   60 D = 0.0E0
      NERR = 1
      GO TO 190
C
C     NORMALIZE ITH ROW BY AIJMAX (JMAX ELEMENT OF ITH ROW)
C
   70 DO  80   J = 1, N
   80 A(I,J) = A(I,J) / AIJMAX
      D = D * AIJMAX
C
C     NORMALIZE ITH ROW OF B
C
      DO  90   J = 1, MD
   90 B(I,J) = B(I,J) / AIJMAX
C
C     USE ROW TRANSFORMATIONS TO GET ZEROS ABOVE AND BELOW THE JMAX
C     ELEMENT OF THE ITH ROW OF A.  APPLY SAME ROW TRANSFORMATIONS
C     TO THE B MATRIX.
C
      DO  120  K = 1, N
      IF   ( K .EQ. I )   GO TO 120
      ARAT = -A(K,JMAX)
      DO  100  J = 1, N
      IF   ( ABS( A(I,J) ) .EQ. 0.0E0 )   GO TO 100
      A(K,J) = ARAT * A(I,J) + A(K,J)
  100 CONTINUE
      A(K,JMAX) = 0.0E0
      DO  110  J = 1, MD
      IF   ( ABS( B(I,J) ) .EQ. 0.0E0 )   GO TO 110
      B(K,J) = ARAT * B(I,J) + B(K,J)
  110 CONTINUE
  120 CONTINUE
```

```
C
C     STORE ROW COUNTER (I) IN TOP ELEMENT OF JMAX COLUMN. THUS,
C     THE TOP ROW OF A WILL CONTAIN THE LOC OF THE PIVOT (UNITY)
C     ELEMENT OF EACH COLUMN (AFTER REDUCTION).
C
      L = I
  130 A(1,JMAX) = FL
C                   THIS STORES INTEGER IN TOP ROW OF A
C
C     THE REDUCTION OF A IS NOW COMPLIETE. PERFORM ROW INTERCHANGES
C     AS INDICATED IN THE FIRST ROW OF A.
C
      DO  170  I = 1, N
      K = I
  140 FK = A(1,K)
C                   THIS PUTS THE INTEGER VALUE IN A INTO K
      IF   ( K - I )140,170,150
C
C     IF K(1,I) IS LESS THAN I, THEN THAT ROW HAS ALREADY BEEN
C     INVOLVED IN AN INTERCHANGE, AND WE USE K(1,K) UNTIL WE GET
C     A VALUE OF K GREATER THAN I  (CORRESPONDING TO A ROW STORED
C     BELOW THE ITH ROW).
C
  150 DO  160  J = 1, MD
      ARAT = B(I,J)
      B(I,J) = B(K,J)
  160 B(K,J) = ARAT
      D = -D
  170 CONTINUE
  180 NERR = 0
  190 RETURN
      END
```

```
      SUBROUTINE MIS2 ( A, N, ND, B, MD, NERR, D )
C
C     MIS2 - COMPLEX INVERT OR SOLVE
C
C     THIS SUBROUTINE SOLVES THE MATRIX EQUATION  A*C = B  WHERE A AND
C          B ARE INPUT.  IF B IS A UNIT MATRIX, THEN C IS THE INVERSF
C          OF A.
C
C          A     = IN-CUT = A TWO DIMENSIONAL COMPLEX ARRAY (DIMENSIONED
C                           ND BY ND) CONTAINING A COMPLEX INPUT MATRIX
C                           CF ORDER N.  THE INPUT MATRIX IS DESTROYED
C                           BY THIS SUBROUTINE.
C
C          N     = INPUT  = THE ORDER (INTEGER) OF A (N LESS THAN OR
C                           EQUAL TO ND) AND THE NUMBER OF ROWS IN B.
C
C          ND    = INPUT  = THE MAXIMUM DIMENSIONS ( INTEGER ) OF THE
C                           SQUARE MATRIX A.
C
C          B     = IN-OUT = A TWO DIMENSIONAL COMPLEX ARRAY (DIMENSIONED
C                           ND BY MD) CONTAINING THE SECOND COMPLEX
C                           MATRIX.  THIS MATRIX IS MODIFIED BY THE
C                           SUBROUTINE AND CONTAINS (UPON OUTPUT) THE C
C                           MATRIX SOLUTION.
C
C          MD    = INPUT  = THE NUMBER (INTEGER) OF COLUMNS IN B.
C
C          NERR  = CUTPUT = OUTPUT CODE ( INTEGER)
C                         = 0   IF A IS NON-SINGULAR
C                         = 1   IF A IS SINGULAR
C                         = 2   IF N IS LESS THAN OR EQUAL TO ZERO
C                         = 3   IF N IS GREATER THAN ND
C
C          D     = INPUT  = COMPLEX SCALE FACTOR FOR DETERMINENT.
C                = CUTPUT = THE SCALED COMPLEX DETERMINENT OF A.
C
      COMPLEX A,AIJMAX,ARAT,B,D,FK,FL
      DIMENSION  A(ND,ND), B(ND,MD)
      EQUIVALENCE  (L,FL), (K,FK)
C
C     CHECK THE VALUE OF N
C
      IF  ( N - 1 )10,20,30
C
   10 D = (0.0E0,0.0E0)
      NERR = 2
      GO TO 190
C
   20 AIJMAX = A(1,1)
      IF  ( CABS(AIJMAX) .EQ. 0.0E0 )  GO TO 60
      B(1,1) = B(1,1) / AIJMAX
      D = D * AIJMAX
```

```
      GO TO 180
C
   30 IF   ( N .LE. ND )   GO TO 40
      NERR = 3
      GO TO 190
C
C     START REDUCTION OF MATRIX A
C
   40 DO  130  I = 1, N
C
C     SEARCH FOR MAXIMUM ELEMENT IN ITH ROW OF A-MATRIX
C
      AIJMAX = A(I,1)
      JMAX = 1
      DO  50  J = 2, N
      IF   ( CABS( A(I,J) ) .LE. CABS( AIJMAX ) )   GO TO 50
      AIJMAX = A(I,J)
      JMAX = J
   50 CONTINUE
C
C     IF AIJMAX IS ZERO, THE MATRIX IS SINGULAR
C
      IF   ( CABS( AIJMAX ) .GT. 0.0E0 )   GO TO 70
   60 D = (0.0E0,0.0E0)
      NERR = 1
      GO TO 190
C
C     NORMALIZE ITH ROW BY AIJMAX (JMAX ELEMENT OF ITH ROW)
C
   70 DO  80  J = 1, N
   80 A(I,J) = A(I,J) / AIJMAX
      D = D * AIJMAX
C
C     NORMALIZE ITH ROW OF B
C
      DO  90  J = 1, MD
   90 B(I,J) = B(I,J) / AIJMAX
C
C     USE ROW TRANSFORMATIONS TO GET ZEROS ABOVE AND BELOW THE JMAX
C     ELEMENT OF THE ITH ROW OF A.  APPLY SAME ROW TRANSFORMATIONS
C     TO THE B MATRIX.
C
      DO  120  K = 1, N
      IF   ( K .EQ. I )   GO TO 120
      ARAT = -A(K,JMAX)
      DO  100  J = 1, N
      IF   ( CABS( A(I,J) ) .EQ. 0.0E0 )   GO TO 100
      A(K,J) = ARAT * A(I,J) + A(K,J)
  100 CONTINUE
      A(K,JMAX) = (0.0E0,0.0E0)
      DO  110  J = 1, MD
      IF   ( CABS( B(I,J) ) .EQ. 0.0E0 )   GO TO 110
```

```
      B(K,J) = ARAT * B(I,J) + B(K,J)
  110 CONTINUE
  120 CONTINUE
C
C     STORE ROW COUNTER (I) IN TOP ELEMENT OF JMAX COLUMN. THUS,
C     THE TOP ROW OF A WILL CONTAIN THE LOC OF THE PIVOT (UNITY)
C     ELEMENT OF EACH COLUMN (AFTER REDUCTION).
C
      L = I
  130 A(1,JMAX) = FL
C                    THIS STORES INTEGER IN TOP ROW OF A
C
C     THE REDUCTION OF A IS NOW COMPLETE. PERFORM ROW INTERCHANGES
C     AS INDICATED IN THE FIRST ROW OF A.
C
      DO  170  I = 1, N
      K = I
  140 FK = A(1,K)
C                    THIS PUTS THE INTEGER VALUE IN A INTO K
      IF   ( K - I )140,170,150
C
C     IF K(1,I) IS LESS THAN I, THEN THAT ROW HAS ALREADY BEEN
C     INVOLVED IN AN INTERCHANGE, AND WE USE K(1,K) UNTIL WE GET
C     A VALUE OF K GREATER THAN I  (CORRESPONDING TO A ROW STORED
C     BELOW THE ITH ROW).
C
  150 DO  160  J = 1, MD
      ARAT = B(I,J)
      B(I,J) = B(K,J)
  160 B(K,J) = ARAT
      D = -D
  170 CONTINUE
  180 NERR = 0
  190 RETURN
      END
```

```
      SUBROUTINE PLT1 (N,X,Y,M)
C
C     PLT1 - LINEAR PLOTTING ROUTINE
C
C     THIS SUBROUTINE GENERATES PRINTED OUTPUT IN GRAPHIC FORM.  BOTH
C         AXES ARE GRADUATED IN A LINEAR MANNER.
C
C                    ***** CDC FTN VERSION OPT=1 *****
C
C         N       = INPUT = NUMBER OF POINTS TO BE PLOTTED (INTEGER)
C
C         X       = INPUT = LINEAR ARRAY (OF LENGTH N) CONTAINING THE
C                           X-COORDINATES
C
C         Y       = INPUT = LINEAR ARRAY (OF LENGTH N) CONTAINING THE
C                           Y-COORDINATES
C
C         M       = INPUT = NUMBER (INTEGER) OF 50 LINE PAGES THE PLOT
C                           OCCUPIES IN Y-DIRECTION
C
C
C     FUNCTIONS USED - ABS, FLOAT
C
C     PRINTED OUTPUT IS PRODUCED ON UNIT 6 AFTER EJECTING A PAGE
C
      DIMENSION FMT(24), FMTT(11), HH(11), FHH(8), FHL(8)
      DIMENSION X(1), Y(1), KJ(1), ME(4), IBUF(101), S(11)
C
      DATA ME(1) / 1H  /, ME(2) / 1H. /,ME(3) / 1H* /,ME(4) / 1HI /
      DATA FMT(1), FMT(24)/4H(8X , 4H)    /,FMTT(1), FMTT(2)/4H(1XA, 4H1
     1  /,FMTT(5), FMTT(6),FMTT(7), FMTT(8),FMTT(11)/4H,1H.,4H,101,4HA1,
     2 ,4H1H. , 4H    )/,HH/4H.0   ,4H.1   ,4H.2   ,4H.3   ,4H.4   ,4H.5   ,
     3 4H.6   ,
     4  4H.7   ,4H,F10,4H,1PE,4H10.2/,FHH/1E8, 1E7, 1E6, 1E5,      1E4, 1
     5E3, 1E2, 1E1/,FHL/1., 1E-1, 1E-2, 1E-3, 1E-4, 1E-5, 1E-6, 1E-7/
C
   10 FORMAT(13X1H.10(9X1H.))
   20 FORMAT (1H1)
   30 FORMAT(1XA1,1PE1C.2,1H.101A1,1H.E10.2)
   40 FORMAT(1XA1,11X101A1)
   50 FORMAT(/33X15A4)
   60 FORMAT  ( 1H1, 29H*** PLT1 ERROR  -  THERE WERE, I6, 32H ARGUMENTS
     1 IN THE CALL STATEMENT / 1H0, 18X, 55HTHERE MUST BE AT LEAST 4 AND
     2 NOT MORE THAN 11 ARGUMENTS )
   70 FORMAT  ( 1H1, 53H*** PLT1 ERROR  -  ALL VALUES OF X AND/OR  Y ARE
     1EQUAL / 1H0, 18X, 6HXMIN =, E12.4, 5X, 6HXMAX =, E12.4, 5X,
     2 6HYMIN =, E12.4, 5X, 6HYMAX =, E12.4 )
C
C
   80 KJ1 = 0
   90 ICODX =  -1
  100 ICODY =  -1
```

```
  110 XMIN   =   X(1)
  120 XMAX   =   X(1)
  130 YMIN   =   Y(1)
  140 YMAX   =   Y(1)
C
C     TEST NUMBER OF POINTS TO PLOT
C
      IF  (N .LT. 2)   RETURN
      L = M
      IF( L .LT. 1) L=1
      XL=L*50
C
C     FIND ACTUAL LIMITS
C
      DO 230 I = 1, N
      IF  (Y (I) .GT.  YMAX)   YMAX = Y (I)
      IF  (Y (I) .LT.  YMIN)   YMIN = Y (I)
      IF  (X (I) .GT.  XMAX)   XMAX = X (I)
  230 IF  (X (I) .LT.  XMIN)   XMIN = X (I)
C
C     TEST FOR XMIN = XMAX   OR  YMIN = YMAX
C
      IF   ( XMIN .EQ. XMAX )   GO TO 490
      IF   ( YMIN .EQ. YMAX )   GO TO 490
C
C     SET UP Y PRINT FORMAT
C
      IF(ICODY .LT.0)   GO  TO  240
      IF(ICODY .GT.7)   GO  TO   240
      J  = ICODY   +  1
      YMI = ABS( YMIN )
      YMX = ABS( YMAX )
      IF   ( YMI .EQ. 0.0 )   YMI = YMX
      IF   ( YMX .EQ. 0.0 )   YMX = YMI
      IF   ( YMI .LT. FHL(J) )   GO TO 240
      IF   ( YMX .LE. FHH(J) )   GO TO 250
C
  240 FMTT(3)      = HH(10)
      FMTT(4)      = HH(11)
      FMTT(9)      = HH(10)
      FMTT(10)     = HH(11)
      GO  TO  260
  250 FMTT ( 3 )   =   HH( 9 )
      FMTT(4)      = HH(J)
      FMTT(9)      = HH(9)
      FMTT(10 )    = HH(J)
C
C     SET UP X PRINT FORMAT
C
  260 IF   ( ICODX .LT. 0 )   GO TO 290
      IF   ( ICODX .GT. 7 )   GO TO 290
      I = ICODX + 1
```

```
      XMI = ABS( XMIN )                                                 PLT1 106
      XMX = ABS( XMAX )                                                 PLT1 107
      IF  ( XMI .EQ. 0.0 )  XMI = XMX                                   PLT1 108
      IF  ( XMX .EQ. 0.0 )  XMX = XMI                                   PLT1 109
      IF  ( XMI .LT. FHL(I) )  GO TO 290                                PLT1 110
      IF  ( XMX .GT. FHH(I) )  GO TO 290                                PLT1 111
C                                                                       PLT1 112
  270 DO  280  K  = 2,  22,  2                                          PLT1 113
      FMT(K)      = HH(9)                                               PLT1 114
  280 FMT(K + 1) = HH(I )                                               PLT1 115
      GO  TO  310                                                       PLT1 116
  290 DO  300  K  = 2,  22,  2                                          PLT1 117
      FMT(K)      = HH(10)                                              PLT1 118
  300 FMT(K + 1) = HH(11)                                               PLT1 119
C                                                                       PLT1 120
  310 YTAG = YMAX - YMIN                                                PLT1 121
      YTAG = YMAX - YMIN                                                PLT1 122
      XTAG=.1*(XMAX-XMIN)                                               PLT1 123
      IF(YTAG.EQ.0.) YTAG=XL                                            PLT1 124
      IF(XTAG.EQ.0.) XTAG=10.                                           PLT1 125
      YFACT=XL/YTAG                                                     PLT1 126
      XFACT=10./XTAG                                                    PLT1 127
      YMAX1=YMIN*YFACT                                                  PLT1 128
      ZMAX1=XMIN*XFACT                                                  PLT1 129
      GPT=1.5-ZMAX1                                                     PLT1 130
      GM=.5/YFACT                                                       PLT1 131
      DO320I=1,11                                                       PLT1 132
  320 S(I)=XMIN+FLOAT(I-1)*XTAG                                         PLT1 133
      IJ=0                                                              PLT1 134
      IF((-ZMAX1).GT.1..AND.(-ZMAX1).LT.99.) IJ=GPT                     PLT1 135
      IV=1                                                              PLT1 136
      LL=50*L+1                                                         PLT1 137
      WRITE (6,20)                                                      PLT1 138
      WRITE (6,FMT ) ( S (I), I  =  1,  11  )                           PLT1 139
      WRITE (6,10)                                                      PLT1 140
      LAST=25*L                                                         PLT1 141
      J=1                                                               PLT1 142
      SY = YMAX                                                         PLT1 143
      DO470LINE=1,LL                                                    PLT1 144
      YSAVE = SY                                                        PLT1 145
      SY=(FLOAT(LL-LINE)+YMAX1)/YFACT                                   PLT1 146
      SZ = SY + GM                                                      PLT1 147
      DO 330 JI=2,100                                                   PLT1 148
  330 IBUF(JI)=ME(1)                                                    PLT1 149
      IF(IJ.GT.0) IBUF(IJ)=ME(2)                                        PLT1 150
      IBUF(1)=ME(2)                                                     PLT1 151
      IBUF(101)=ME(2)                                                   PLT1 152
      DO 370  I = 1, N                                                  PLT1 153
      IF  (Y (I) .GE. YSAVE .OR. Y (I) .LT. SY)  GO TO 370              PLT1 154
      JJ = X (I) * XFACT + GPT                                          PLT1 155
      IF(IBUF(JJ).EQ.ME(4)) GO TO 370                                   PLT1 156
      JI=3                                                              PLT1 157
```

```
      IF (Y (I) .LT. SZ)  JI = 2
      IF(IBUF(JJ).EQ.ME(1)) GO TO 360
      IF(IBUF(JJ).EQ.ME(2)) GO TO 340
      IF(JI.EQ.3) GO TO 370
      GOTO350
340   IF(JI.EQ.2) GOTO370
350   IBUF(JJ)=ME(4)
      GOTO370
360   IBUF(JJ)=ME(JI)
370   CONTINUE
380   JJ=3
      II=2
      IF (LINE .NE. 1)  GO TO 390
      IF (SY .LE. 0.)  IV = 0
      GO TO 400
390   IF (LINE .EQ. LL)  GO TO 400
      IF(SY.GT.0..OR.IV.EQ.0)  GO TO 430
      IV=0
      IF (SZ .LE. 0.)  GO TO 400
      JJ=2
      II=3
400   DO420 JI=1,101
      IF(IBUF(JI).EQ.ME(JJ)) GOTO410
      IBUF(JI)=ME(II)
      GOTO420
410   IBUF(JI)=ME(4)
420   CONTINUE
430   IF ( KJ1.EQ.0.OR.J.GE.61.OR.(LAST-LINE).GT.30)  GO TO 440
      K=KJ(J)
      J=J+1
      IF  ( L .NE. 1 )  GO TO 450
      IF  ( J .NE. 52 )  GO TO 450
440   K=ME(1)
450   IF (MOD (LINE,5) .NE. 1)  GO TO 460
      WRITE (5,FMTT)K, SY, (IBUF (I), I = 1, 101), SY
      GOTO470
460   WRITE (5,40)  K, (IBUF (I), I = 1, 101)
470   CONTINUE
      WRITE (6,10)
      WRITE (5,FMT) (S (I), I = 1, 11)
      IF ( KJ1.NE.0 )  WRITE (6,50)  ( KJ(J), J = 61, 75 )
      RETURN
C
C     PRINT ERROR  -  THERE WERE AN INVALID NUMBER OF ARGUMENTS
C
480   WRITE  (6,60)   NPAR
      RETURN
C
C     PRINT ERROR  -  XMIN = XMAX  OR  YMIN = YMAX
C
490   WRITE  (6,70)   XMIN, XMAX, YMIN, YMAX
      RETURN
```

PLT1 210

END

```
      SUBROUTINE PRNT  (A,M,N,MAXR,NRC,TITLE1,NTITLE)
C     PRNT - MATRIX PRINT SUBROUTINE
C            MATRIX MAY BE - REAL OR COMPLEX
C                          - SINGLE OR DOUBLY DIMENSIONED
C     A      IS THE MATRIX TO BE PRINTED
C     M      IS THE NUMBER OF ROWS TO BE PRINTED
C     N      IS THE NUMBER OF COLUMNS TO BE PRINTED
C   MAXR  IS THE MAXIMUM DIMENSIONED ROWS IN THE CONTROL PROGRAM
C    KROW IS THE MAXIMUM NUMBER OF DATA ROWS TO BE PRINTED ON EACH
C              PAGE.(MAXIMUM OF 61), FOR 8.5 BY 11 PAGE USE 36)
C     NRC    IS AN INDICATOR FOR REAL OR COMPLEX
C       NRC = 1 (REAL FLOATING)
C       NRC = 2 (COMPLEX FLOATING)
      COMMON /ZZZ/CASE(30),TITLE(18),NIN,NOUT,KROW,LINES,IPRNT,NER
      DIMENSION A(1),TITLE1(1)
   10 FORMAT(1H ,3X,I4,8E14.5)
   20 FORMAT(//   2X,8(6X,I8))
   30 FORMAT (1H+ 65X,18HHAS ZERO DIMENSION)
   40 FORMAT (/1H0,3X,6A10)
      IF (IPRNT .LE. 0) GO TO 140
      IF (M*N)50,50,60
   50 WRITE (NOUT   ,40) (TITLE(I),I=1,12)
      WRITE (NOUT   ,30)
      GO TO 140
   60 MAXR2 = MAXR * NRC
      JB = 1
      JB2 = 1
      JB3=8/NRC
      IF(KROW .EQ. 0) KROW = 61
      IF(KROW .EQ. 47 .AND. NRC.EQ.1) JB3=6
      IF(JB3 .GE. N) JB3 = N
      JE =(JB3-1)* MAXR2+ 1
      KROW1 = KROW - 11
      JMAX = (N-1)* MAXR2 + NRC* M - NRC/2
      IM = 0
      IF(LINES.EQ.0) GO TO 70
      IF(LINES.LT.(KROW-10)) GO TO 80
   70 CALL HEADNG
      LINES = 2
   80 IF (NTITLE .EQ. 0) GO TO 90
      NTITL = (NTITLE*4+9)/10
      WRITE (NOUT,40) (TITLE1(I),I=1,NTITL)
      GO TO 100
   90 WRITE (NOUT   ,40) (TITLE(I),I=1,12)
  100 LINES = LINES + 3
  110 IF (NRC.EQ.1)WRITE(NOUT   ,20) (J, J=JB2,JB3)
      IF (NRC.EQ.2)WRITE(NOUT   ,20) (J,J, J = JB2,JB3)
      LINES = LINES + 3
  120 IM = IM + 1
      IF (NRC.EQ.1) WRITE(NOUT   ,10) IM,(A(J), J= JB,JE,MAXR2)
      IF (NRC.EQ.2) WRITE(NOUT   ,10) IM,(A(J),A(J+1), J=JB,JE,MAXR2)
      JB = JB + NRC
```

```
      JE = JE + NRC
      LINES = LINES + 1
      IF (IM .GE. M) GO TO 130
      IF (LINES .GE. KROW) GO TO 70
      GO TO 120
  130 IF (JE .GT. JMAX) RETURN
      JB = JB3 * MAXR2 + 1
      JB2 = JB3 + 1
      JB3=JB3+8/NRC
      IF (KROW .EQ. 47 .AND. NRC.EQ.1)  JB3 = JB3 - 1
      IF (JB3 .GE. N) JB3 = N
      IM = 0
      JE = (JB3-1)*MAXR2 + 1
      IF (LINES .LT. KROW1) GO TO 110
      GO TO 70
  140 RETURN
      END
```

```
      SUBROUTINE RDTAPE (NT,A,IA,NOUT,NER)                      RDTAPE  2
      DIMENSION A(IA)                                           RDTAPE  3
      READ (NT) A                                               RDTAPE  4
      IF (EOF(NT).NE.0) NER = 51                                RDTAPE  5
   30 RETURN                                                    RDTAPE  6
      END                                                       RDTAPE  7
```

```
      SUBROUTINE TRAERO (NOUT,NER,MTOTAP,NTOTAP                         TRAERO 2
     1 , CR,D,F,WORK,SPLS,SPLA)                                         TRAERO 3
C                                                                       TRAERO 4
      DIMENSION CR(3,20)                                                TRAERO 5
      DIMENSION D(1),WORK(1),F(MTOTAP,1)                                TRAERO 6
      DIMENSION SPLS(NTOTAP,1),SPLA(NTOTAP,1)                           TRAERO 7
      COMMON/DISK2/ND2,ITBL2(843),NRECSA,IBUMP,NKD,VOBWS(20)            DISK2  2
      COMMON /XTF/NF(100)                                               TRAERO 9
      EQUIVALENCE (NF(6),NSYM),       (NF(7),NASYM)                     TRAERO10
      EQUIVALENCE (NF(41),NK),        (NF(42),NG)                       TRAERO11
     1           ,(NF(43),NBOX),      (NF(44),NSBETO),   (NF(45),NB)    TRAERO12
      DIMENSION GUST(6), AERO(6)                                        TRAERO13
      DATA GUST/4HFPSP,4HFZSP,4HFYSP,4HFPAP,4HFZAP,4HFYAP/              TRAERO14
      DATA AERO/4HDPSG,4HDZSG,4HDYSG,4HDPAG,4HDZAG,4HDYAG/              TRAERO15
C                                                                       TRAERO16
      NRECSA=2*(NKD*(1+NG)+1)+3                                         TRAERO17
      CALL OPENMS(ND2,ITBL2,NRECSA,0)                                   TRAERO18
      IBUMP=2*(1+NG)                                                    TRAERO19
C                                                                       TRAERO20
      LOC=NKD*IBUMP+3                                                   TRAERO21
      CALL RDAERO (4HGEOL,0,WORK,NOUT,NER)                              TRAERO22
      CALL WRITMS (ND2,WORK,5*NBOX,LOC)                                 TRAERO23
      LOC=LOC+1                                                         TRAERO24
      CALL RDAERO (4HGEOS,0,WORK,NOUT,NER)                              TRAERO25
      CALL WRITMS (ND2,WORK,5*NSBETO,LOC)                               TRAERO26
C                                                                       TRAERO27
      CALL RDAERO (4HCR   ,0,CR,NOUT,NER)                               TRAERO28
      IF (NER.NE.0) GO TO 1000                                          TRAERO29
      CALL RDAERO (4HRK   ,0,VOBWS,NOUT,NER)                            TRAERO30
      IF (NER.NE.0) GO TO 1000                                          TRAERO31
      DO   35 I=1,NKD                                                   TRAERO32
      IF (VOBWS(I).EQ.0.0) VOBWS(I) = 1.0E-10                           TRAERO33
   35 VOBWS(I) = 1.0/VOBWS(I)                                           TRAERO34
C                                                                       TRAERO35
C     NOW SAVE SPLS,SPLA                                                TRAERO36
C                                                                       TRAERO37
      CALL RDAERO (4HHPS ,0,D,NOUT,NER)                                 TRAERO38
      N = 0                                                             TRAERO39
      DO 210 J=1,NSYM                                                   TRAERO40
      DO 210 I=1,NBOX                                                   TRAERO41
      N = N+1                                                           TRAERO42
  210 SPLS(I,J) = D(N)                                                  TRAERO43
      IF (NASYM.EQ.0) GO TO 220                                         TRAERO44
      CALL RDAERO (4HHPA ,0,D,NOUT,NER)                                 TRAERO45
      N = 0                                                             TRAERO46
      DO 215 J=1,NASYM                                                  TRAERO47
      DO 215 I=1,NBOX                                                   TRAERO48
      N = N+1                                                           TRAERO49
  215 SPLA(I,J) = D(N)                                                  TRAERO50
  220 IF (NB.EQ.0) GO TO 260                                            TRAERO51
      CALL RDAERO (4HHZS ,0,D,NOUT,NER)                                 TRAERO52
      I1 = NBOX+1                                                       TRAERO53
```

```
      I2 = NBOX+NSBETO                                                  TRAER054
      N = 0                                                             TRAER055
      DO 230 J=1,NSYM                                                   TRAER056
      DO 230 I=I1,I2                                                    TRAER057
      N = N+1                                                           TRAER058
  230 SPLS(I,J) = D(N)                                                  TRAER059
      IF (NASYM.EQ.0) GO TO 240                                         TRAER060
      CALL RDAERO (4HHZA ,0,D,NOUT,NER)                                 TRAER061
      N = 0                                                             TRAER062
      DO 235 J=1,NASYM                                                  TRAER063
      DO 235 I=I1,I2                                                    TRAER064
      N = N+1                                                           TRAER065
  235 SPLA(I,J) = D(N)                                                  TRAER066
  240 CALL RDAERO (4HHYS ,0,D,NOUT,NER)                                 TRAER067
      I1 = I2+1                                                         TRAER068
      I2 = I1+NSBETO-1                                                  TRAER069
      N = 0                                                             TRAER070
      DO 250 J=1,NSYM                                                   TRAER071
      DO 250 I=I1,I2                                                    TRAER072
      N = N+1                                                           TRAER073
  250 SPLS(I,J) = D(N)                                                  TRAER074
      IF (NASYM.EQ.0) GO TO 260                                         TRAER075
      CALL RDAERO (4HHYA ,0,D,NOUT,NER)                                 TRAER076
      N = 0                                                             TRAER077
      DO 255 J=1,NASYM                                                  TRAER078
      DO 255 I=I1,I2                                                    TRAER079
      N = N+1                                                           TRAER080
  255 SPLA(I,J) = D(N)                                                  TRAER081
C                                                                       TRAER082
  260 NW=NTOTAP*NSYM                                                    TRAER083
      LOC=NKD*IBUMP+1                                                   TRAER084
      CALL WRITMS(ND2,SPLS,NW,LOC)                                      TRAER085
      NW=NTOTAP*NASYM                                                   TRAER086
      LOC=LOC+1                                                         TRAER087
      CALL WRITMS(ND2,SPLA,NW,LOC)                                      TRAER088
C                                                                       TRAER089
      DO 200 K=1,NKD                                                    TRAER090
C                                                                       TRAER091
      KBUMP=(K-1)*IBUMP                                                 TRAER092
      LOCO=KBUMP                                                        TRAER093
      M=2                                                               TRAER094
      IG = 0                                                            TRAER095
C                                                                       TRAER096
  110 NAERO = 2*NBOX                                                    TRAER097
      I2=0                                                              TRAER098
      M=M+1                                                             TRAER099
      DO 140 MB=1,3                                                     TRAER100
      IG = IG+1                                                         TRAER101
      I1 = I2+1                                                         TRAER102
      I2 = I1+NAERO-1                                                   TRAER103
      IF (MB.GT.1.AND.NB.EQ.0) GO TO 140                                TRAER104
C     READ GUST DATA                                                    TRAER105
```

```
      CALL RDAERO (GUST(IG),K,WORK,NOUT,NER)
C
      N = 0
      DO 130 J=1,NG
       DO 130 I=I1,I2
      N = N+1
  130 F(I,J) = WORK(N)
C
      NAERO = 2*NSBETO
  140 CONTINUE
C
      DO 135 J=1,NG
      LOC=LOCO+M+2*(J-1)
  135 CALL WRITMS(ND2,F(1,J),MTOTAP,LOC)
C
      IF (NASYM.EQ.0) GO TO 150
      IF(M.GT.3)GO TO 150
      GO TO 110
C
C     GENERALIZED AERO
C
  150 NM = NSYM
      IND = 1
      M=1
      IA = 1
C
  160 NWD = 2*NM*NM
      CALL RDAERO (AERO(IA),K,D,NOUT,NER)
      IF (NB.EQ.0) GO TO 100
      DO 90 N=1,2
      IA = IA+1
      CALL RDAERO (AERO(IA),K,WORK,NOUT,NER)
      DO 75 I=1,NWD
   75 D(I)    = D(I)    + WORK(I)
   90 CONTINUE
C
  100 LOC=M+KBUMP
      CALL WRITMS(ND2,D,NWD,LOC)
C
      IF (NASYM.EQ.0) GO TO 200
      IF (IND.EQ.3) GO TO 200
      NM = NASYM
      IND = 3
      M=2
      IA = 4
      GO TO 160
C
  200 CONTINUE
C
 1000 RETURN
      END
```

```
      SUBROUTINE WRAERO (ITYPR,IK,IHD,NOUT,NER)                         WRAERO 2
      DIMENSION IHD(50)                                                 WRAERO 3
C                                                                       WRAERO 4
      COMMON /DDTBLS/DDTBL(20,10)                                       DDTBLS 2
      INTEGER DDTBL                                                     DDTBLS 3
      LOGICAL READ                                                      WRAERO 6
C                                                                       WRAERO 7
      DIMENSION TYPE(31)                                                WRAERO 8
      DATA TYPE/4HHEAD,4HGEOM,4HGEO1,4HGEO3,4HGEOS,4HCR  ,4HRK          WRAERO 9
     1     ,     4HHPS ,4HHPA ,4HHZS ,4HHZA ,4HHYS ,4HHYA               WRAERO10
     2     ,     4HDPSP,4HFPSP,4HDZSP,4HFZSP,4HDYSP,4HFYSP              WRAERO11
     3     ,     4HDPAP,4HFPAP,4HDZAP,4HFZAP,4HDYAP,4HFYAP              WRAERO12
     4     ,     4HDPSG,4HDZSG,4HDYSG,4HDPAG,4HDZAG,4HDYAG/             WRAERO13
      DATA ITYP/4HAERO/                                                 WRAERO14
C                                                                       WRAERO15
      READ = .FALSE.                                                    WRAERO16
      GO TO 10                                                          WRAERO17
C                                                                       WRAERO18
      ENTRY RDAERO                                                      WRAERO19
      READ = .TRUE.                                                     WRAERO20
C                                                                       WRAERO21
   10 NER = 0                                                           WRAERO22
      IF (J.NE.0) GO TO 20                                              WRAERO23
      CALL FDDTBL(ITYP,J,JO)                                            WRAERO24
      IF (J.NE.0) GO TO 20                                              WRAERO25
      NER = 54                                                          WRAERO26
      GO TO 1000                                                        WRAERO27
C                                                                       WRAERO28
   20 NT = DDTBL(1,J)                                                   WRAERO29
      IF (NT.EQ.0) GO TO 1000                                           WRAERO30
      IF (DDTBL(20,J).EQ.0) GO TO 50                                    WRAERO31
      WRITE (NOUT,1) ITYP                                               WRAERO32
    1 FORMAT (1H0,A4,* FILE ERROR CONDITION - FILE CANNOT BE PROCESSED*)WRAERO33
      GO TO 1000                                                        WRAERO34
C                                                                       WRAERO35
   50 DO 60 I=1,31                                                      WRAERO36
      ITYPR = I                                                         WRAERO37
      IF (TYPR.EQ.TYPE(I)) GO TO 90                                     WRAERO38
   60 CONTINUE                                                          WRAERO39
      GO TO 900                                                         WRAERO40
C                                                                       WRAERO41
   90 NSYM  = DDTBL(11,J)                                               WRAERO42
      NS = 2                                                            WRAERO43
      IF (NASYM.EQ.0) NS=1                                              WRAERO44
      NASYM = DDTBL(12,J)                                               WRAERO45
      NGUST = DDTBL(13,J)                                               WRAERO46
      NAGUST = NGUST                                                    WRAERO47
      IF (NASYM.EQ.0) NAGUST=0                                          WRAERO48
      NB = 3                                                            WRAERO49
      IF (DDTBL(14,J).EQ.0) NB=1                                        WRAERO50
      IF (NB.EQ.3) GO TO 95                                             WRAERO51
C                                                                       WRAERO52
```

```
      IF (ITYPR.EQ.5) GO TO 900                                         WRAER053
      IF (ITYPR.GT.9.AND.ITYPR.LT.14) GO TO 900                         WRAER054
      IF (ITYPR.GT.15.AND.ITYPR.LT.20) GO TO 900                        WRAER055
      IF (ITYPR.GT.21.AND.ITYPR.LT.26) GO TO 900                        WRAER056
      IF (ITYPR.EQ.27.OR.ITYPR.EQ.28) GO TO 900                         WRAER057
      IF (ITYPR.EQ.30.OR.ITYPR.EQ.31) GO TO 900                         WRAER058
C                                                                       WRAER059
   95 IF (NASYM.NE.0) GO TO 100                                         WRAER060
      IF (ITYPR.EQ.9.OR.ITYPR.EQ.11.OR.ITYPR.EQ.13) GO TO 900           WRAER061
      IF (ITYPR.GT.19.AND.ITYPR.LT.26) GO TO 900                        WRAER062
      IF (ITYPR.GT.28.AND.ITYPR.LT.32) GO TO 900                        WRAER063
C                                                                       WRAER064
  100 IF (ITYPR.GT.7) GO TO 150                                         WRAER065
C                                                                       WRAER066
      IRN = ITYPR                                                       WRAER067
      NREAD = 1                                                         WRAER068
      GO TO (101,102,103,104,105,106,107), ITYPR                        WRAER069
C                                                                       WRAER070
  101 NW = 50                                                           WRAER071
      GO TO 200                                                         WRAER072
  102 NW = DDTBL(15,J)*3 + DDTBL(17,J)*8 + DDTBL(18,J)                  WRAER073
     1   + DDTBL(19,J)*2 + DDTBL(14,J)*3                                WRAER074
      GO TO 200                                                         WRAER075
  103 NW = DDTBL(15,J)*6                                                WRAER076
      GO TO 200                                                         WRAER077
  104 NW = DDTBL(15,J)*3                                                WRAER078
      GO TO 200                                                         WRAER079
  105 NW = DDTBL(16,J)*6                                                WRAER080
      GO TO 200                                                         WRAER081
  106 NW = 3*DDTBL(13,J)                                                WRAER082
      IF (NB.EQ.1) IRN = IRN-1                                          WRAER083
      GO TO 200                                                         WRAER084
  107 NW = DDTBL(10,J)                                                  WRAER085
      IF (NB.EQ.1) IRN = IRN-1                                          WRAER086
      GO TO 200                                                         WRAER087
C                                                                       WRAER088
  150 IRN = 7                                                           WRAER089
      IF (NB.EQ.3) IRN = IRN+1                                          WRAER090
      IF (ITYPR.GT.13) GO TO 180                                        WRAER091
C          INTERPOLATION MATRICES                                       WRAER092
C                                                                       WRAER093
       NW = DDTBL(15,J)                                                 WRAER094
       NREAD = NSYM                                                     WRAER095
       IF (ITYPR.EQ.8) GO TO 200                                        WRAER096
      IRN = IRN+NSYM                                                    WRAER097
       NREAD = NASYM                                                    WRAER098
       IF (ITYPR.EQ.9) GO TO 200                                        WRAER099
       NW = DDTBL(16,J)                                                 WRAER100
      IRN = IRN+NASYM                                                   WRAER101
       NREAD = NSYM                                                     WRAER102
       IF (ITYPR.EQ.10) GO TO 200                                       WRAER103
      IRN = IRN+NSYM                                                    WRAER104
```

```
      NREAD = NASYM                                                     WRAER105
       IF (ITYPR.EQ.11) GO TO 200                                       WRAER106
      IRN = IRN+NASYM                                                   WRAER107
       NREAD = NSYM                                                     WRAER108
       IF (ITYPR.EQ.12) GO TO 200                                       WRAER109
      IRN = IRN+NSYM                                                    WRAER110
       NREAD = NASYM                                                    WRAER111
      GO TO 200                                                         WRAER112
C                                                                       WRAER113
  180 IRN = IRN + (NSYM+NASYM)*NB                                       WRAER114
      IRN = IRN + (IK-1)*NB*(NSYM+NGUST+NASYM+NAGUST+NS)                WRAER115
      IF (ITYPR.GT.25) GO TO 190                                        WRAER116
C           PHYSICAL FORCES                                             WRAER117
C                                                                       WRAER118
      NW = DDTBL(15,J)*2                                                WRAER119
      NREAD = NSYM                                                      WRAER120
      IF (ITYPR.EQ.14) GO TO 200                                        WRAER121
      IRN = IRN+NSYM                                                    WRAER122
      NREAD = NGUST                                                     WRAER123
      IF (ITYPR.EQ.15) GO TO 200                                        WRAER124
      NW = DDTBL(16,J)*2                                                WRAER125
      IRN = IRN+NGUST                                                   WRAER126
      NREAD = NSYM                                                      WRAER127
      IF (ITYPR.EQ.16) GO TO 200                                        WRAER128
      IRN = IRN+NSYM                                                    WRAER129
      NREAD = NGUST                                                     WRAER130
      IF (ITYPR.EQ.17) GO TO 200                                        WRAER131
      IRN = IRN+NGUST                                                   WRAER132
      NREAD = NSYM                                                      WRAER133
      IF (ITYPR.EQ.18) GO TO 200                                        WRAER134
      IRN = IRN+NSYM                                                    WRAER135
      NREAD = NGUST                                                     WRAER136
      IF (ITYPR.EQ.19) GO TO 200                                        WRAER137
      NW = DDTBL(15,J)*2                                                WRAER138
      IRN = IRN+NGUST                                                   WRAER139
      NREAD = NASYM                                                     WRAER140
      IF (ITYPR.EQ.20) GO TO 200                                        WRAER141
      IRN = IRN+NASYM                                                   WRAER142
      NREAD = NGUST                                                     WRAER143
      IF (ITYPR.EQ.21) GO TO 200                                        WRAER144
      NW = DDTBL(16,J)*2                                                WRAER145
      IRN = IRN+NGUST                                                   WRAER146
      NREAD = NASYM                                                     WRAER147
      IF (ITYPR.EQ.22) GO TO 200                                        WRAER148
      IRN = IRN+NASYM                                                   WRAER149
      NREAD = NGUST                                                     WRAER150
      IF (ITYPR.EQ.23) GO TO 200                                        WRAER151
      IRN = IRN+NGUST                                                   WRAER152
      NREAD = NASYM                                                     WRAER153
      IF (ITYPR.EQ.24) GO TO 200                                        WRAER154
      IRN = IRN+NASYM                                                   WRAER155
      NREAD = NGUST                                                     WRAER156
```

```
      GO TO 200
C
  190 IRN = IRN + NB*(NSYM+NGUST+NASYM+NAGUST)
C           GENERALIZED FORCES
C
      IRN = IRN+ITYPR-26
      NREAD = 1
      NW = NSYM*(NSYM+NGUST)
      IF (ITYPR.LT.29) GO TO 200
      NW = NASYM*(NASYM+NAGUST)
      GO TO 200
C
  200 IF (READ) GO TO 250
      IF (DDTBL(4,J).EQ.0) GO TO 210
      NER = 1
      GO TO 1000
C
  210 IRN = DDTBL(5,J) + 1
      IF (ITYPR.EQ.1) IHD(1) = ITYP
      CALL WRTAPE (NT,IHD,NW,NOUT,NER)
      IF (NER.NE.0) GO TO 990
      GO TO 500
C
  250 DDTBL(4,J) = 1
      LRN = DDTBL(5,J)
      NSKIP = IRN-LRN-1
      IF (NSKIP.EQ.0) GO TO 300
      IF (NSKIP.GT.0) GO TO 280
      REWIND NT
      WRITE (NOUT,9) NT
    9 FORMAT (1H0,5(4H****)/5HOTAPE,I2,8H REWOUND/1H0,5(4H****) )
      IF (ITYPR.EQ.1) GO TO 300
      NSKIP = IRN-1
C
  280 DO 290 I=1,NSKIP
  290 READ(NT)
C
  300 IRN = IRN-1
      N = 1
      DO 310 I=1,NREAD
      CALL RDTAPE (NT,IHD(N),NW,NOUT,NER)
      IF (NER.NE.0) GO TO 990
      IRN = IRN+1
  310 N = N+NW
C
  500 DDTBL(5,J) = IRN
      IF (ITYPR.NE.1) GO TO 1000
C
      IF (IHD(1).EQ.ITYP) GO TO 510
      NER = 1
      WRITE (NOUT,501) ITYP, IHD(1)
  501 FORMAT (1H0,*INVALID FILE*/1H0,*FILE SHOULD BE =*,A4
```

```
    1 /1H ,*FILE IS          =*,A4)
      GO TO 990
C
  510 DO 520 I=2,11
  520 DDTBL(I+8,J) = IHD(I)
C
      IF (.NOT.READ) WRITE (NOUT,3) NT
    3 FORMAT (28H HEADER DATA WRITTEN ON TAPE,I2)
      IF (READ) WRITE (NOUT,4) NT
    4 FORMAT (28H HEADER DATA READ  FROM TAPE,I2)
      WRITE (NOUT,5)   (IHD(I),I=2,11), (IHD(I),I=21,25)
    5 FORMAT
     1 (8X2HNK,5X4HNSYM,5X5HNASYM,5X5HNGUST,8X2HNB,6X4HNBOX,6X4HNSBE
     2 , 4X6HNSTRIP,4X6HMAXSTR,8X2HNP /10I10/
     3 / 6X4HMACH,6X4HREFA,6X4HREFS,6X4HREFC, 8X2HXM /5F10.2 )
      GO TO 1000
C
  900 NER = 1
      WRITE (NOUT,2) TYPR
    2 FORMAT (1HO,*INVALID REQUEST FROM AERO FILE FOR RECORD TYPE=*,A4)
C
  990 DDTBL(20,J) = 1
C
 1000 RETURN
      END
```

```
      SUBROUTINE WRFRSP (IR,IG,IK,IHD,NOUT,NER)                         WRFRSP 2
      DIMENSION IHD(50)                                                 WRFRSP 3
      COMMON /DDTBLS/DDTBL(20,10)                                       DDTBLS 2
      INTEGER DDTBL                                                     DDTBLS 3
      LOGICAL READ                                                      WRFRSP 5
      DATA ITYP/4HFRSP/                                                 WRFRSP 6
C                                                                       WRFRSP 7
      READ = .FALSE.                                                    WRFRSP 8
      GO TO 10                                                          WRFRSP 9
      ENTRY RDFRSP                                                      WRFRSP10
      READ = .TRUE.                                                     WRFRSP11
C                                                                       WRFRSP12
   10 IF (J.NE.0) GO TO 20                                              WRFRSP13
      CALL FDDTBL(ITYP,JI,J)                                            WRFRSP14
      IF (J.NE.0) GO TO 20                                              WRFRSP15
      NER = 1                                                           WRFRSP16
      GO TO 1000                                                        WRFRSP17
C                                                                       WRFRSP18
   20 NT = DDTBL(1,J)                                                   WRFRSP19
      IF (NT.EQ.0) GO TO 1000                                           WRFRSP20
      IF (DDTBL(20,J).EQ.0) GO TO 50                                    WRFRSP21
      WRITE (NOUT,1) ITYP                                               WRFRSP22
    1 FORMAT (1H0,A4,  FILE ERROR CONDITION - FILE CANNOT BE PROCESSED )WRFRSP23
      NER = 1                                                           WRFRSP24
      GO TO 1000                                                        WRFRSP25
C                                                                       WRFRSP26
   50 IF (IR.GT.0.AND.IR.LT.10) GO TO 100                               WRFRSP27
      NER = 1                                                           WRFRSP28
      GO TO 1000                                                        WRFRSP29
C                                                                       WRFRSP30
  100 IF (IR.EQ.1) NW=50                                                WRFRSP31
      IF (IR.EQ.2) NW = DDTBL(11,J)*3                                   WRFRSP32
      IF (IR.EQ.3) NW = DDTBL(12,J)                                     WRFRSP33
      IF(IR.EQ.4)NW=5                                                   WRFRSP34
      IF (IR.EQ.5) NW = DDTBL(15,J)                                     WRFRSP35
      IF (IR.EQ.6) NW = DDTBL(15,J)                                     WRFRSP36
      IF (IR.EQ.7) NW = 2*(DDTBL(13,J)+DDTBL(14,J) )                    WRFRSP37
      IF (IR.EQ.8) NW = 2*DDTBL(16,J)                                   WRFRSP38
      IF (IR.EQ.9) NW = 2*DDTBL(16,J)                                   WRFRSP39
C                                                                       WRFRSP40
      IF (READ) GO TO 200                                               WRFRSP41
      IF (DDTBL(4,J).EQ.0) GO TO 110                                    WRFRSP42
      NER = 1                                                           WRFRSP43
      GO TO 1000                                                        WRFRSP44
C                                                                       WRFRSP45
  110 IRN = DDTBL(5,J)+1                                                WRFRSP46
      IF (IR.EQ.1) IHD(1) = ITYP                                        WRFRSP47
C                                                                       WRFRSP48
  150 CALL WRTAPE (NT,IHD,NW,NOUT,NER)                                  WRFRSP49
      GO TO 500                                                         WRFRSP50
C                                                                       WRFRSP51
  200 IRN = IR                                                          WRFRSP52
```

```
      IF (IR.LT.4) GO TO 250                                           WRFRSP53
      IRN = IR + 6*(IK-1 + DDTBL(12,J)*(IG-1) )                        WRFRSP54
C                                                                      WRFRSP55
  250 LRN = DDTBL(5,J)                                                 WRFRSP56
      NRSKP = IRN-LRN-1                                                WRFRSP57
      IF (NRSKP.EQ.0) GO TO 300                                        WRFRSP58
      IF (NRSKP.GT.0) GO TO 280                                        WRFRSP59
      REWIND NT                                                        WRFRSP60
      WRITE (NOUT,2) NT                                                WRFRSP61
    2 FORMAT (1H0,5(4H****)/5HOTAPE,I2,8H REWOUND/1H0,5(4H****) )      WRFRSP62
      IF (IR.EQ.1) GO TO 300                                           WRFRSP63
      NRSKP = IRN-1                                                    WRFRSP64
C                                                                      WRFRSP65
  280 DO 290 I=1,NRSKP                                                 WRFRSP66
  290 READ(NT)                                                         WRFRSP67
C                                                                      WRFRSP68
  300 CALL RDTAPE (NT,IHD,NW,NOUT,NER)                                 WRFRSP69
      IF (NER.NE.0) GO TO 1000                                         WRFRSP70
      DDTBL(4,J) = 1                                                   WRFRSP71
C                                                                      WRFRSP72
  500 IF (IR.GT.1) GO TO 550                                           WRFRSP73
      IF (IHD(1).EQ.ITYP) GO TO 510                                    WRFRSP74
      NER = 1                                                          WRFRSP75
      WRITE (NOUT,501) ITYP, IHD(1)                                    WRFRSP76
  501 FORMAT (1H0, INVALID FILE /1H0, FILE SHOULD BE = ,A4             WRFRSP77
     1 /1H , FILE IS        = ,A4)                                     WRFRSP78
      GO TO 990                                                        WRFRSP79
C                                                                      WRFRSP80
  510 DO 520 I=3,8                                                     WRFRSP81
  520 DDTBL(I+8,J) = IHD(I)                                            WRFRSP82
      IF (.NOT.READ) WRITE (NOUT,3) NT                                 WRFRSP83
      IF (READ) WRITE (NOUT,4) NT                                      WRFRSP84
      WRITE (NOUT,5) (IHD(I),I=2,7), (IHD(I),I=11,12)                  WRFRSP85
    3 FORMAT (28H HEADER DATA WRITTEN ON TAPE,I2)                      WRFRSP86
    4 FORMAT (28H HEADER DATA READ  FROM TAPE,I2)                      WRFRSP87
    5 FORMAT (5X5HIDENT,5X5HNGUST,5X5HNFREQ,6X4HNSYM,5X5HNASYM,7X3HNKI WRFRSP88
     2 , 7X3HVEL,5X,5HSIGMA                                            WRFRSP89
     3 / 6I10,2F10.2)                                                  WRFRSP90
C                                                                      WRFRSP91
  550 DDTBL(5,J) = IRN                                                 WRFRSP92
      GO TO 1000                                                       WRFRSP93
C                                                                      WRFRSP94
  990 DDTBL(20,J) = 1                                                  WRFRSP95
C                                                                      WRFRSP96
 1000 RETURN                                                           WRFRSP97
      END                                                              WRFRSP98
```

```
      SUBROUTINE WRLOAD (IR,IG,IHD,NOUT,NER)                            WRLOAD 2
      DIMENSION IHD(50)                                                 WRLOAD 3
      COMMON /DDTBLS/DDTBL(20,10)                                       DDTBLS 2
      INTEGER DDTBL                                                     DDTBLS 3
      LOGICAL READ                                                      WRLOAD 5
      DATA ITYP/4HLOAD/                                                 WRLOAD 6
C                                                                       WRLOAD 7
      READ = .FALSE.                                                    WRLOAD 8
      GO TO 10                                                          WRLOAD 9
      ENTRY RDLOAD                                                      WRLOAD10
      READ = .TRUE.                                                     WRLOAD11
C                                                                       WRLOAD12
   10 IF (J.NE.0) GO TO 20                                              WRLOAD13
      CALL FDDTBL(ITYP,JI,J)                                            WRLOAD14
      IF (J.NE.0) GO TO 20                                              WRLOAD15
      NER = 1                                                           WRLOAD16
      GO TO 1000                                                        WRLOAD17
C                                                                       WRLOAD18
   20 NT = DDTBL(1,J)                                                   WRLOAD19
      IF (NT.EQ.0) GO TO 1000                                           WRLOAD20
      IF (DDTBL(20,J).EQ.0) GO TO 50                                    WRLOAD21
      WRITE (NOUT,1) ITYP                                               WRLOAD22
    1 FORMAT (1H0,A4,  FILE ERROR CONDITION - FILE CANNOT BE PROCESSED )WRLOAD23
      NER = 1                                                           WRLOAD24
      GO TO 1000                                                        WRLOAD25
C                                                                       WRLOAD26
   50 IF (IR.GT.0.AND.IR.LT.10) GO TO 100                               WRLOAD27
      NER = 1                                                           WRLOAD28
      GO TO 1000                                                        WRLOAD29
C                                                                       WRLOAD30
  100 CONTINUE                                                          WRLOAD31
      IF (IR.EQ.1) NW=50                                                WRLOAD32
      IF (IR.EQ.2) NW = DDTBL(13,J)*8                                   WRLOAD33
      IF (IR.EQ.3) NW = 3*DDTBL(11,J)                                   WRLOAD34
      IF (IR.EQ.4) NW = DDTBL(12,J)                                     WRLOAD35
      IF (IR.EQ.5) NW = 2*DDTBL(14,J)                                   WRLOAD36
      IF (IR.EQ.6) NW = 2*DDTBL(14,J)*DDTBL(12,J)                       WRLOAD37
      IF (IR.EQ.7) NW = 2*DDTBL(14,J)*DDTBL(12,J)                       WRLOAD38
      IF (IR.EQ.8) NW = 2*DDTBL(13,J)*DDTBL(12,J)                       WRLOAD39
      IF (IR.EQ.9) NW = 2*DDTBL(13,J)*DDTBL(12,J)                       WRLOAD40
C                                                                       WRLOAD41
      IF (READ) GO TO 200                                               WRLOAD42
      IF (DDTBL(4,J).EQ.0) GO TO 110                                    WRLOAD43
      NER = 1                                                           WRLOAD44
      GO TO 1000                                                        WRLOAD45
C                                                                       WRLOAD46
  110 IRN = DDTBL(5,J)+1                                                WRLOAD47
      IF (IR.EQ.1) IHD(1) = ITYP                                        WRLOAD48
      CALL WRTAPE (NT,IHD,NW,NOUT,NER)                                  WRLOAD49
      GO TO 500                                                         WRLOAD50
C                                                                       WRLOAD51
  200 IRN = IR                                                          WRLOAD52
```

```
      IF (IR.LT.5) GC TC 250                                            WRLOAD53
      M = 4                                                             WRLOAD54
      IF (DDTBL(14,J).NE.0) GO TO 220                                   WRLOAD55
      IRN = IRN-3                                                       WRLOAD56
      M = 2                                                             WRLOAD57
      IF (IR.EQ.8.OR.IR.EQ.9) GO TO 220                                 WRLOAD58
      NER = 201                                                         WRLOAD59
      WRITE (NOUT,6) TYPR                                               WRLOAD60
    6 FORMAT (1HO,*INVALID REQUEST FROM AERO FILE FOR RECORD TYPE=*,A4) WRLOAD61
      GO TO 990                                                         WRLOAD62
C                                                                       WRLOAD63
  220 IRN = IRN + M*(IG-1)                                              WRLOAD64
C                                                                       WRLOAD65
  250 LRN = DDTBL(5,J)                                                  WRLOAD66
      NRSKP = IRN-LRN-1                                                 WRLOAD67
      IF (NRSKP.EQ.0) GO TO 300                                         WRLOAD68
      IF (NRSKP.GT.0) GO TO 280                                         WRLOAD69
      REWIND NT                                                         WRLOAD70
      WRITE (NOUT,2) NT                                                 WRLOAD71
    2 FORMAT (1HO,5(4H****)/5HOTAPE,I2,8H REWOUND/1HO,5(4H****) )       WRLOAD72
      IF (IR.EQ.1) GC TC 300                                            WRLOAD73
      NRSKP = IRN-1                                                     WRLOAD74
C                                                                       WRLOAD75
  280 DO 290 I=1,NRSKP                                                  WRLOAD76
  290 READ(NT)                                                          WRLOAD77
C                                                                       WRLOAD78
  300 CALL RDTAPE (NT,IHD;NW,NOUT,NER)                                  WRLOAD79
      IF (NER.NE.0) GO TO 1000                                          WRLOAD80
      DDTBL(4,J) = 1                                                    WRLOAD81
C                                                                       WRLOAD82
  500 IF (IR.GT.1) GO TO 550                                            WRLOAD83
      IF (IHD(1).EQ.ITYP) GO TO 510                                     WRLOAD84
      NER = 1                                                           WRLOAD85
      WRITE (NOUT,501) ITYP, IHD(1)                                     WRLOAD86
  501 FORMAT (1HO, INVALID FILE /1HO, FILE SHOULD BE = ,A4              WRLOAD87
     1 /1H , FILE IS         = ,A4)                                     WRLOAD88
      GO TO 990                                                         WRLOAD89
C                                                                       WRLOAD90
  510 DO 520 I=3,6                                                      WRLOAD91
  520 DDTBL(I+8,J) = IHD(I)                                             WRLOAD92
      IF (.NOT.READ) WRITE (NOUT,3) NT                                  WRLOAD93
      IF (READ) WRITE (NOUT,4) NT                                       WRLOAD94
      WRITE (NOUT,5) (IHD(I),I=2,6), (IHD(I),I=11,12)                   WRLOAD95
    3 FORMAT (28H HEADER DATA WRITTEN ON TAPE,I2)                       WRLOAD96
    4 FORMAT (28H HEADER DATA READ  FROM TAPE,I2)                       WRLOAD97
    5 FORMAT (5X5HIDENT,5X5HNGUST,5X5HNFREQ,4X6HNINTLD,6X4HNACC         WRLOAD98
     1 , 7X3HVEL,5X5HSIGMA                                              WRLOAD99
     2 /5I10,2F10.2)                                                    WRLOA100
C                                                                       WRLOA101
  550 DDTBL(5,J) = IRN                                                  WRLOA102
      GO TO 1000                                                        WRLOA103
C                                                                       WRLOA104
```

```
  990 DDTBL(20,J) = 1                                                    WRLOA105
C                                                                        WRLOA106
 1000 RETURN                                                             WRLOA107
      END                                                                WRLOA108
```

```
      SUBROUTINE WRTAPE (NT,A,IA,NOUT,NER)                              WRTAPE 2
      DIMENSION A(IA)                                                   WRTAPE 3
      WRITE (NT) A                                                      WRTAPE 4
      RETURN                                                            WRTAPE 5
      END                                                               WRTAPE 6
```

```
      SUBROUTINE WRUNIT (IR,IK,IHD,NOUT,NER)                                WRUNIT 2
      DIMENSION IHD(50)                                                     WRUNIT 3
      COMMON /DDTBLS/DDTBL(20,10)                                           DDTBLS 2
      INTEGER DDTBL                                                         DDTBLS 3
      LOGICAL READ                                                          WRUNIT 5
      DATA ITYP/4HUNIT/                                                     WRUNIT 6
C                                                                           WRUNIT 7
      READ = .FALSE.                                                        WRUNIT 8
      GO TO 10                                                              WRUNIT 9
      ENTRY RDUNIT                                                          WRUNIT10
      READ = .TRUE.                                                         WRUNIT11
C                                                                           WRUNIT12
   10 IF (J.NE.0) GO TO 20                                                  WRUNIT13
      CALL FDDTBL(ITYP,JI,J)                                                WRUNIT14
      IF (J.NE.0) GO TO 20                                                  WRUNIT15
      NER = 1                                                               WRUNIT16
      GO TO 1000                                                            WRUNIT17
C                                                                           WRUNIT18
   20 NT = DDTBL(1,J)                                                       WRUNIT19
      IF (NT.EQ.0) GO TO 1000                                               WRUNIT20
      IF (DDTBL(20,J).EQ.0) GO TO 50                                        WRUNIT21
      WRITE (NOUT,1) ITYP                                                   WRUNIT22
    1 FORMAT (1H0,A4,  FILE ERROR CONDITION - FILE CANNOT BE PROCESSED )WRUNIT23
      NER = 1                                                               WRUNIT24
      GO TO 1000                                                            WRUNIT25
C                                                                           WRUNIT26
   50 IF (IR.GT.0.AND.IR.LT.16) GO TO 100                                   WRUNIT27
      NER = 1                                                               WRUNIT28
      GO TO 1000                                                            WRUNIT29
C                                                                           WRUNIT30
  100 CONTINUE                                                              WRUNIT31
      IF (IR.EQ.1) NW=50                                                    WRUNIT32
      IF (IR.EQ.2) NW = DDTBL(10,J)*8                                       WRUNIT33
      IF (IR.EQ.3) NW = DDTBL(11,J)*DDTBL(10,J)                             WRUNIT34
      IF (IR.EQ.4) NW = DDTBL(10,J)*DDTBL(12,J)                             WRUNIT35
      IF (IR.EQ.5) NW = DDTBL(13,J)*DDTBL(12,J)                             WRUNIT36
      IF (IR.EQ.6) NW = DDTBL(10,J)*(DDTBL(13,J)+DDTBL(14,J))               WRUNIT37
      IF (IR.EQ.7) NW = 2*DDTBL(10,J)*DDTBL(13,J)                           WRUNIT38
      IF (IR.EQ.8) NW = 2*DDTBL(10,J)*DDTBL(14,J)                           WRUNIT39
      IF (IR.EQ.9) NW = DDTBL(10,J)                                         WRUNIT40
      IF (IR.EQ.10) NW = DDTBL(10,J)                                        WRUNIT41
      IF (IR.EQ.11) NW = 3*DDTBL(16,J)                                      WRUNIT42
      IF (IR.EQ.12) NW = DDTBL(10,J)*DDTBL(16,J)*DDTBL(18,J)                WRUNIT43
      IF (IR.EQ.13) NW = 3*DDTBL(17,J)                                      WRUNIT44
      IF (IR.EQ.14.OR.IR.EQ.15) NW = DDTBL(10,J)*DDTBL(17,J)*DDTBL(19,J) WRUNIT45
C                                                                           WRUNIT46
      IF (READ) GO TO 200                                                   WRUNIT47
      IF (DDTBL(4,J).EQ.0) GO TO 110                                        WRUNIT48
      NER = 1                                                               WRUNIT49
      GO TO 990                                                             WRUNIT50
C                                                                           WRUNIT51
  110 IRN = DDTBL(5,J)+1                                                    WRUNIT52
```

```
      IF (IR.EQ.1) IHD(1) = ITYP                                        WRUNIT53
      CALL WRTAPE (NT,IHD,NW,NOUT,NER)                                  WRUNIT54
      GO TO 500                                                         WRUNIT55
C                                                                       WRUNIT56
  200 IRN = IR                                                          WRUNIT57
      IF (IR.LT.7) GO TO 250                                            WRUNIT58
      IF (IR.GT.8) GO TO 210                                            WRUNIT59
      IRN = IR+2*(IK-1)                                                 WRUNIT60
      GO TO 250                                                         WRUNIT61
C                                                                       WRUNIT62
  210 IRN = IR-2 + 2*DDTBL(15,J)                                        WRUNIT63
C                                                                       WRUNIT64
  250 LRN = DDTBL(5,J)                                                  WRUNIT65
      NRSKP = IRN-LRN-1                                                 WRUNIT66
      IF (NRSKP.EQ.0) GO TO 300                                         WRUNIT67
      IF (NRSKP.GT.0) GO TO 280                                         WRUNIT68
      REWIND NT                                                         WRUNIT69
      WRITE (NOUT,2) NT                                                 WRUNIT70
    2 FORMAT (1H0,5(4H****)/5HOTAPE,I2,8H REWOUND/1H0,5(4H****) )       WRUNIT71
      IF (IR.EQ.1) GO TO 300                                            WRUNIT72
      NRSKP = IRN-1                                                     WRUNIT73
C                                                                       WRUNIT74
  280 DO 290 I=1,NRSKP                                                  WRUNIT75
  290 READ(NT)                                                          WRUNIT76
C                                                                       WRUNIT77
  300 CALL RDTAPE (NT,IHD,NW,NOUT,NER)                                  WRUNIT78
      IF (NER.NE.0) GO TO 1000                                          WRUNIT79
      DDTBL(4,J) = 1                                                    WRUNIT80
C                                                                       WRUNIT81
  500 IF (IR.GT.1) GO TO 550                                            WRUNIT82
      IF (IHD(1).EQ.ITYP) GO TO 510                                     WRUNIT83
      NER = 1                                                           WRUNIT84
      WRITE (NOUT,501) ITYP, IHD(1)                                     WRUNIT85
  501 FORMAT (1H0, INVALID FILE /1H0, FILE SHOULD BE = ,A4              WRUNIT86
     1 /1H , FILE IS          = ,A4)                                    WRUNIT87
      GO TO 990                                                         WRUNIT88
C                                                                       WRUNIT89
  510 DO 520 I=3,12                                                     WRUNIT90
  520 DDTBL(I+7,J) = IHD(I)                                             WRUNIT91
      IF (.NOT.READ) WRITE (NOUT,3) NT                                  WRUNIT92
      IF (READ) WRITE (NOUT,4) NT                                       WRUNIT93
      WRITE (NOUT,5) (IHD(I),I=2,12)                                    WRUNIT94
    3 FORMAT (28H HEADER DATA WRITTEN ON TAPE,I2)                       WRUNIT95
    4 FORMAT (28H HEADER DATA READ  FROM TAPE,I2)                       WRUNIT96
    5 FORMAT (5X5HIDENT,4X6HNINTLD,4X6HNSTRSS,5X5HNENGS,6X4HNSYM        WRUNIT97
     1 , 5X5HNASYM,8X2HNK                                               WRUNIT98
     2 , 4X6HNABGRP,4X6HNSBGRP,4X6HNBOXES,4X6HNAERSB                    WRUNIT99
     3 / 11I10)                                                         WRUNI100
C                                                                       WRUNI101
  550 DDTBL(5,J) = IRN                                                  WRUNI102
      GO TO 1000                                                        WRUNI103
C                                                                       WRUNI104
```

```
  990 DDTBL(20,J) = 1                                                    WRUNI105
C                                                                        WRUNI106
 1000 RETURN                                                             WRUNI107
      END                                                                WRUNI108
```

```
      SUBROUTINE CSDLM (INPUT)                                          CSDLM   2
C                                                                       CSDLM   3
C        DUMMY MAIN FOR THE DOUBLET-LATTICE PART OF THE VIBRA-4 PROGRAMCSDLM   4
C                                                                       CSDLM   5
      COMMON /AROCOM/ NTI, MODES                                        AROCOM 2
     X , NP, MSTRIP, NSMAX, NCMAX, NTOTAL, NB, MSBE, MBE               AROCOM 3
     Y , ND, NE, NBY, NBZ, NTU, NTP, NTY, NTZ                           AROCOM 4
     1 , NTYS, NTZS, MAXGR, MAXSTR, NSBETO, NSTRIP, KR, XM, RLFA, REFC  AROCOM 5
     2 , REFS, FMACH, LINES                                             AROCOM 6
C                                                                       CSDLM   7
      COMMON NAA,A(1)                                                   CSDLM   8
C                                                                       CSDLM   9
   10 FORMAT (6I12)                                                     CSDLM  10
C                                                                       CSDLM  11
      NTI = INPUT                                                       CSDLM  12
      DATA SECT/4HSECT/, AERO/4HAERO/                                   CSDLM  13
C                                                                       CSDLM  14
C     FIND THE SECT OR AERO INPUT DATA                                  CSDLM  15
C                                                                       CSDLM  16
      ISECT = 0                                                         CSDLM  17
      READ (NTI,90) CHECK                                               CSDLM  18
   90 FORMAT (A4)                                                       CSDLM  19
      IF (CHECK.NE.SECT) GO TO 110                                      CSDLM  20
C     SECT INPUT DATA FOUND                                             CSDLM  21
      ISECT = 1                                                         CSDLM  22
      READ (NTI,10) NMS, NDOF, NSYM, NASYM                              CSDLM  23
C                                                                       CSDLM  24
  100 READ (NTI,90) CHECK                                               CSDLM  25
  110 IF (CHECK.NE.AERO) GO TO 100                                      CSDLM  26
C                                                                       CSDLM  27
C     AERO INPUT DATA FOUND                                             CSDLM  28
C                                                                       CSDLM  29
      READ (NTI,10) NODES, NSYM, NASYM, MFIX1, MFIX2                    CSDLM  30
      MDOFP = NODES                                                     CSDLM  31
      MDOFB = NODES                                                     CSDLM  32
      IF (ISECT.NE.0) MDOFP = 4*NODES                                   CSDLM  33
      IF (ISECT.NE.0) MDOFB = 2*NODES                                   CSDLM  34
      MODES = NSYM + NASYM                                              CSDLM  35
      READ (NTI,10) NP, MSTRIP, NSMAX, NCMAX, NBOXES                    CSDLM  36
      NSMAX = NSMAX+1                                                   CSDLM  37
      NCMAX = NCMAX+1                                                   CSDLM  38
      READ (NTI,10) NB, MSBE, MBE                                       CSDLM  39
      NTOTAL = NBOXES + 2*MAX0(MSBE, MBE)                               CSDLM  40
      MSBE = MSBE+1                                                     CSDLM  41
      MBE = MBE+1                                                       CSDLM  42
C                                                                       CSDLM  43
      L1  = NAA                                                         CSDLM  44
      L2  = L1    +  NODES                                              CSDLM  45
      L3  = L2    +  NODES                                              CSDLM  46
      L4  = L3    +  NODES                                              CSDLM  47
      L5  = L4    +  MDOFP * MODES                                      CSDLM  48
      L6  = L5    +  MDOFB * MODES                                      CSDLM  49
```

```
      L7  = L6    +  MDOF3 * MODES                                  CSDLM 50
      L8  = L7    +  NP                                             CSDLM 51
      L9  = L8    +  NP * NB                                        CSDLM 52
      L10 = L9    +  NP                                             CSDLM 53
      L11 = L10   +  NP                                             CSDLM 54
      L12 = L11   +  NP                                             CSDLM 55
      L13 = L12   +  MSTRIP                                         CSDLM 56
      L14 = L13   +  MSTRIP                                         CSDLM 57
      L15 = L14   +  2*NB                                           CSDLM 58
      L16 = L15   +  2*NB                                           CSDLM 59
      L17 = L16   +  2*NB                                           CSDLM 60
      L18 = L17   +  NB                                             CSDLM 61
      L19 = L18   +  NB                                             CSDLM 62
      L20 = L19   +  NB                                             CSDLM 63
      L21 = L20   +  NB                                             CSDLM 64
      L22 = L21   +  NB                                             CSDLM 65
      L23 = L22   +  NB                                             CSDLM 66
      L24 = L23   +  NB                                             CSDLM 67
      L25 = L24   +  MBE                                            CSDLM 68
      L27 = L25 + NB*10                                             CSDLM 69
      L28 = L27   +  MSBE                                           CSDLM 70
      L29 = L28   +  MSBE                                           CSDLM 71
      L30 = L29   +  MSBE                                           CSDLM 72
      L31 = L30   +  MSBE                                           CSDLM 73
      L32 = L31   +  MSTRIP + NB                                    CSDLM 74
      L33 = L32   +  MSTRIP + NB                                    CSDLM 75
      L34 = L33   +  MSTRIP + NB                                    CSDLM 76
      L35 = L34   +  MSTRIP + NB                                    CSDLM 77
      L36 = L35   +  MSTRIP + NB                                    CSDLM 78
      L37 = L36   +  MSTRIP + NB                                    CSDLM 79
      L38 = L37   +  MSTRIP + NB                                    CSDLM 80
      L39 = L38   +  MSTRIP + NB                                    CSDLM 81
      L40 = L39   +  MSTRIP + NB                                    CSDLM 82
      L41 = L40   +  MSTRIP + NB                                    CSDLM 83
      L42 = L41   +  NTOTAL                                         CSDLM 84
      L43 = L42   +  NTOTAL                                         CSDLM 85
      L44 = L43   +  NTOTAL                                         CSDLM 86
      L45 = L44   +  NTOTAL                                         CSDLM 87
      L46 = L45   +  NTOTAL                                         CSDLM 88
      L47 = L46   +  NTOTAL                                         CSDLM 89
      L47A = L47   + NTOTAL                                         CSDLM 90
      L48   = L47A + 4*NTOTAL                                       CSDLM 91
      L49 = L48 + 2*NTOTAL                                          CSDLM 92
      L50 = L49 + 2*NTOTAL                                          CSDLM 93
      L51 = L50 + 2*(MAX0(NSYM,NASYM)+20)                           CSDLM 94
      NAA = L51                                                     CSDLM 95
      IF (ISECT.EQ.0) GO TO 200                                     CSDLM 96
      ISECT = L51                                                   CSDLM 97
      NAA = L48 + NDOF*NMS*MODES                                    CSDLM 98
C                                                                   CSDLM 99
      IPR = 1                                                       CSDLM100
      CALL INPUTA (NTI,NMS,NDOF,MODES,IPR                           CSDLM101
```

```
     1 , A(L1), A(L2), A(L3), A(L48) )                                  CSDLM102
C                                                                       CSDLM103
 200 CONTINUE                                                           CSDLM104
C                                                                       CSDLM105
C     CALL       SDLM (ELXI,ELYI,ELZI,PHIN,PHIZ,PHIY, NODES             CSDLM106
C    1 , NSYM, NASYM, MFIX1, MFIX2, ISECT, NDOF, PHI, MDOFP, MDOFB      CSDLM107
C    2 , NAS, NASB, NSARAY, NCARAY, NSARAY, ISSTR, NSSTR                CSDLM108
C    3 , IFLA, NBEA, NT12, NSBEA                                        CSDLM109
C    4 , YB, ZB, ARB, AVR, XLE, XTE, RIA, THIA, AO, AOP, XIS1, XIS2     CSDLM110
C    5 , CG, CS, EE, SG, YS, ZS, XIJ, YIN, ZIN, COORD                   CSDLM111
C    6 , X, XIC, DELX, XLAM, H, DHDX, DELA                              CSDLM112
C    7 , DT, DTA, RHS )                                                 CSDLM113
C                                                                       CSDLM114
      CALL SDLM (A(L1), A(L2), A(L3), A(L4), A(L5), A(L6), NODES        CSDLM115
     1 , NSYM, NASYM, MFIX1, MFIX2, ISECT, NDOF, A(L48), MDOFP, MDOFB   CSDLM116
     2 , A(L7), A(L8), A(L9), A(L10), A(L11), A(L12), A(L13)            CSDLM117
     3 , A(L14), A(L15), A(L16), A(L17)                                 CSDLM118
     4 , A(L18), A(L19), A(L20), A(L21), A(L22), A(L23)                 CSDLM119
     4 , A(L24), A(L25), A(L27), A(L28), A(L29), A(L30)                 CSDLM120
     5 , A(L31), A(L32), A(L33), A(L34), A(L35), A(L36), A(L37), A(L38) CSDLM121
     5 , A(L39), A(L40)                                                 CSDLM122
     6 , A(L41), A(L42), A(L43), A(L44), A(L45), A(L46), A(L47)         CSDLM123
     7 , A(L48), A(L49), A(L50), A(L47A) )                              CSDLM124
C                                                                       CSDLM125
      RETURN                                                            CSDLM126
      END                                                               CSDLM127
```

```
      SUBROUTINE  AERO (MFIX1,MFIX2,NG,NMSYM,NTOT,NOUT,NW,NEWBFZ,         AERO   2
     1         NEWBFY, NSTCT, CBSPAN,DCP , FZ , FY , CN, CM, SPLD         AERO   3
     2 , ISSTR,NSBEA,NBARAY,NCARAY,YB,ZB,XIS1,XIS2,CG,CS,EE,SG,YS,ZS      AERO   4
     3 , XIC, XIJ, DELX, COORD, CZB, CYB, CNB, CMB, CPR, CPI )            AERO   5
      DIMENSION ISSTR(1),NSBEA(1),NBARAY(1),NCARAY(1),YB(1),ZB(1)         AERO   6
      DIMENSION XIS1(1),XIS2(1),CG(1),CS(1),EE(1),SG(1),YS(1),ZS(1)       AERO   7
      DIMENSION XIC(1),XIJ(1),DELX(1),COORD(1)                            AERO   8
C   ***   COMPUTES  AERODYNAMIC PARAMETERS FOR ALL  LIFTING SURFACES      AERO   9
C         AND ALL SLENDER BODIES                                          AERO  10
      COMMON /AROCOM/ NTI, MODES                                          AROCOM 2
     X , NP, MSTRIP, NSMAX, NCMAX, NTOTAL, NB, MSBE, MBE                  AROCOM 3
     Y , ND, NE, NBY, NBZ, NTO, NTP, NTY, NTZ                             AROCOM 4
     1 , NTYS, NTZS, MAXGR, MAXSTR, NSBETO, NSTRIP, KR, XM, REFA, REFC    AROCOM 5
     2 , REFS, FMACH, LINES                                               AROCOM 6
      COMPLEX DCP(1), FZ(1), FY(1), CN(1), CM(1), SPLD(1)                 AERO  12
      COMPLEX CZB(1), CYB(1), CNB(1), CMB(1)                              AERO  13
      DIMENSION CPR(1), CPI(1)                                            AERO  14
      COMPLEX CZT, CYT, CMT, CNT, CXT, CLT, FZLB, FYLB                    AERO  15
      COMPLEX    CZTS, CYTS, CMTS, CNTS, CLTS                             AERO  16
      REAL       KR                                                       AERO  17
   10 FORMAT  ( 1H1 /// 10X, 21H***    PRESSURES    *** //                AERO  18
     1          13X, 4HKR = , F8.4 / )                                    AERO  19
   20 FORMAT  ( // 45X, 7HMODE NO,I4//4X,8HPANEL    , 10H  STRIP    ,     AERO  20
     1        10H     BOX    ,7X,3HXOC,8X,1HX,11X,1HY,11X,1HZ,15X,        AERO  21
     2        10H PRESSURES/ )                                            AERO  22
   30 FORMAT  ( 1H1 /// 10X, 21HDEFINITION OF SYMBOLS    //  10X,         AERO  23
     1        18HS      =   SEMI-SPAN/10X, 25HC-BAR =  REFERENCE CHORD /) AERO  24
   40 FORMAT  ( 10X, 20HL      =   BODY LENGTH / 10X, 45HF-Z/Q =  (VERTICAAERO  25
     1L FORCE)/(DYNAMIC PRESSURE)           / 10X, 45HF-Y/Q =  (LATERALAERO  26
     2  FORCE)/(DYNAMIC PRESSURE)     // )                                AERO  27
   50 FORMAT  (1H1//41X,4HMODE, I4 // 7H  STRIP, 5X, 1HY,   9X, 1HZ,      AERO  28
     1     5X,3HY/S,10X,16HLIFT COEFFICIENT, 7X,18HMOMENT COEFFICIENT,    AERO  29
     2    6X, 18HCENTER OF PRESSURE / 64X, 18H1/4 CHORD OF  STRIP   /     AERO  30
     3     41X, 18HREAL          IMAG  , 5X, 18HREAL           IMAG. ,    AERO  31
     4      9X, 14HREAL       IMAG.     / )                               AERO  32
   60 FORMAT  ( 1H1/// 41X, 4HMODE, I4 // 6H  BODY, 6X,1HY,11X,1HZ,5X,    AERO  33
     1  34X/L,4X,49HRUNNING LOAD (VERTICAL)   RUNNING LOAD (LATERAL)   /  AERO  34
     2  7H  ELEM., 36X, 38H(DF-Z/Q) / DX            (DF-Y/Q) / DX      /  AERO  35
     3          39X, 45HREAL             IMAG.        REAL          IMAG./) AERO  36
   70 FORMAT  (  6X, I4, 6X,I4,6X, I4,3X, 4F12.5,2F12.6 )                 AERO  37
   72 FORMAT  ( I5, 3F10.4, 4F12.6, 2F10.4)                               AERO  38
   80 FORMAT  ( // 10X, 15HSIGN CONVENTION  //  10X,                      AERO  39
     1        39HTHE FOLLOWING TWO SIGN CONVENTIONS HOLD  //  10X,        AERO  40
     2        76H(1)  FORCES  AND DEFLECTIONS POSITIVE DOWN ON HORIZONTALAERO  41
     3  SURFACES OR Z-BODIES   / 15X, 54HAND POSITIVE OUTBOARD ON VERTICALAERO  42
     4  SURFACES OR Y-BODIES   //10X,                                     AERO  43
     5        76H(2)  FORCES  AND DEFLECTIONS POSITIVE  UP  ON HORIZONTALAERO  44
     6  SURFACES OR Z-BODIES   / 15X, 54HAND POSITIVE INBOARD  ON VERTICALAERO  45
     7  SURFACES OR Y-BODIES   //    )                                    AERO  46
   90 FORMAT  (/// 25X, 14HTOTALS ON BODY , I3   // 10X, 7HF-Z/Q = ,      AERO  47
     1        2F15.6, 10X, 7HM-Z/Q = , 2F15.6    / 10X, 7HF-Y/Q = ,       AERO  48
     2        2F15.6, 10X, 7HM-Y/Q = , 2F15.6    / )                      AERO  49
```

```
100  FORMAT  ( ///35X, 4HMODE,I3//25X,26HTOTALS ON LIFTING SURFACES )
110 FORMAT  ( /10X, 7HCZ     =, 2F15.6, 10X, 7HCY     =, 2F15.6 / 10X,
   1           7HCM     =, 2F15.6, 10X, 7HCN     =, 2F15.6 / 10X,
   2           7HCSL    =, 2F15.6 / )
120 FORMAT  (// 25X, 27HTOTALS ON ENTIRE  AIRCRAFT      )
130 FORMAT  ( I5, F12.4, 3X, 4E14.5 )
140 FORMAT  ( 1H1 )
150 FORMAT  ( // 48H  STATION  SPANWISE    SPAN LOAD  (C*CL)/(C-BAR)
   1          /  46H           COORD.          REAL            IMAG. /)
      WRITE  (NOUT,10)  KR
C
      REWIND  NW
      DO  180  K = 1, MFIX2
      READ  (NW)   (DCP(IK), IK=1,NTCT)
      IF ( K.NE.MFIX1 .AND. K.NE.MFIX2 ) GO TO 180
      KK = K
      IF (K.GT.NMSYM) KK = K-NG
      WRITE (NOUT,20) KK
      I2          =  0
      LP          =  1
      LPAGE       =  1
      DO  170   J = 1, NSTRIP
      I1          =   I2 + 1
      I2          =   I2 + NCARAY(LP)
      DO  160  I = I1, I2
      XCC   =  (XIC(I)-XIJ(J)) / CS(J)
      WRITE   (NOUT,70)   LP, J, I, XCC, XIC(I), YS(J), ZS(J), DCP(I)
      IF   (LPAGE*LINES .NE. I)  GO TO  160
      LPAGE       =   LPAGE + 1
      WRITE (NOUT,20) KK
160 CONTINUE
      IF  (I2.EQ.NBARAY(LP))   LP=LP+1
170 CONTINUE
180 CONTINUE
C
C    ***
C
      SYMA    = 2.0
      SYMB    = 0.0
      DO 190 I=1,NSTCT
      CPR(I) = 0.0
      CPI(I) = 0.0
      FZ(I) = (0.0,0.0)
      FY(I) = (0.0,0.0)
190 CONTINUE
      WRITE   (NOUT,30)
      IF   (NB . EQ . 0)   GO TO  210
      WRITE   (NOUT,40)
      REWIND   NEWBFZ
      REWIND   NEWBFY
210 CONTINUE
      REWIND   NW
```

```
      WRITE  (NOUT,80)                                                  AERO 102
      DO  390  K=1, MFIX2                                               AERO 103
      READ  (NW)   (DCP(I),  I=1,NTOT)                                  AERO 104
      IF  (K .NE. MFIX1 .AND. K .NE. MFIX2)  GO TO  271                 AERO 105
      I2    = 0                                                         AERO 106
      LP    = 1                                                         AERO 107
      KK = K                                                            AERO 108
      IF (K.GT.NMSYM) KK = K-NG                                         AERO 109
      WRITE (NOUT,50) KK                                                AERO 110
      IF  (K .LE. NMSYM)  GO TO  212                                    AERO 111
      SYMA    = 0.0                                                     AERO 112
      SYMB    =   2.0                                                   AERO 113
  212 CONTINUE                                                          AERO 114
      DO  260  J=1,NSTRIP                                               AERO 115
      I1    = I2+1                                                      AERO 116
      I2    = I2+NCARAY(LP)                                             AERO 117
      IF  (I2.EQ.NBARAY(LP))  LP=LP+1                                   AERO 118
      XI14S = 0.25*CS(J) + XIJ(J)                                       AERO 119
      CN(J)= (0.0,0.0)                                                  AERO 120
      CM(J)= (0.0,0.0)                                                  AERO 121
      SPLD(J)   =   0.0                                                 AERO 122
      DO  220  I=I1,I2                                                  AERO 123
      CN(J)= CN(J) + DCP(I)*DELX(I)                                     AERO 124
      CM(J)= CM(J) + DCP(I)*DELX(I)*(XIC(I) - XI14S)                    AERO 125
  220 CONTINUE                                                          AERO 126
      YOS   = YS(J)/REFS                                                AERO 127
      CH2      = CS(J)**2                                               AERO 128
      JL         =   ISSTR(J)                                           AERO 129
      SPLD(JL) = SPLD(JL) + CN(J) / CBSPAN                              AERO 130
      CN(J)   = CN(J)/CS(J)                                             AERO 131
      CM(J)   =-CM(J)/CH2                                               AERO 132
      ABSCNJ = SQRT((REAL(CN(J)))**2 + (AIMAG(CN(J)))**2)               AERO 133
      IF   (ABS(REAL(CN(J))) .LE. 0.0000001)  GO TO  222                AERO 134
      IF  (ABSCNJ.LE.0.00001) GO TO 230                                 AERO 135
      CPR(J) =-REAL(CM(J))/REAL(CN(J)) + 0.25                           AERO 136
      GO TO  224                                                        AERO 137
  222 CPR(J) = 0.0                                                      AERO 138
  224 CONTINUE                                                          AERO 139
      IF   (ABS(AIMAG(CN(J))) .LE. 0.0000001)  GO TO  240               AERO 140
      CPI(J) =-AIMAG(CM(J))/AIMAG(CN(J))+0.25                           AERO 141
      GO TO  250                                                        AERO 142
  230 CONTINUE                                                          AERO 143
      CPR(J) = 0.0                                                      AERO 144
  240 CONTINUE                                                          AERO 145
      CPI(J) = 0.0                                                      AERO 146
  250 CONTINUE                                                          AERO 147
      WRITE   (NOUT, 72)  J,YS(J), ZS(J), YOS,CN(J),CM(J),CPR(J),CPI(J) AERO 148
  260 CONTINUE                                                          AERO 149
      WRITE   (NOUT,140)                                                AERO 150
      WRITE   (NOUT,150)                                                AERO 151
      DO  270   JL = 1, MAXSTR                                          AERO 152
      WRITE   (NOUT,130)   JL, COORD(JL), SPLD(JL)                      AERO 153
```

```
270 CONTINUE
271 CONTINUE
    IF (NTZS.EQ.0. AND .NTYS.EQ.0)  GO TO  300
C
    READ    (NEWBFZ)   (FZ(LL),              LL=1, NSBETO)
    READ    (NEWBFY)   (FY(LL),              LL=1, NSBETO)
    IF   (K .NE. MFIX1 .AND. K .NE. MFIX2)  GO TO  390
    L2    = 0
    DO  290  N=1, NB
    CZB(N) = (0.0,0.0)
    CYB(N) = (0.0,0.0)
    CMB(N) = (0.0,0.0)
    CNB(N) = (0.0,0.0)
    L1    = L2+1
    L2    = L2 + NSBEA(N)
    SBL     =  XIS2(L2) - XIS1(L1)
    WRITE  (NOUT,60) KK
    DO   280  LB=L1,L2
    LX      = LX+1
    XSB   = 0.5*(XIS1(LB) + XIS2(LB))
    DXSB  = XIS2(LB) - XIS1(LB)
    XOL     = (XSB -XIS1(L1))/ SBL
    FZLB = FZ(LB)/ DXSB
    FYLB = FY(LB)/ DXSB
    WRITE   (NOUT, 72)  LB, YB(N), ZB(N), XOL, FZLB, FYLB
    CZB(N) = CZB(N) + FZ(LB)
    CYB(N) = CYB(N) + FY(LB)
    CMB(N) = CMB(N) - FZ(LB)*(XSB-XIS1(L1))
    CNB(N) = CNB(N) - FY(LB)*(XSB-XIS1(L1))
280 CONTINUE
    CZB(N) = CZB(N)/REFA
    CYB(N) = CYB(N)/REFA
    CMB(N) = CMB(N)/(REFA*REFC)
    CNB(N) = CNB(N)/(REFA*REFC)
    XCEN    = (XIS1(L1) + XIS2(L2))/2.0
    WRITE   (NOUT, 90)   N, CZB(N), CMB(N), CYB(N), CNB(N)
290 CONTINUE
300 CONTINUE
    WRITE   (NOUT,140)
    CZT   = (0.0,0.0)
    CYT   = (0.0,0.0)
    CMT   = (0.0,0.0)
    CNT   = (0.0,0.0)
    CLT   = (0.0,0.0)
    DO   320  J=1,NSTRIP
    CH2   = CS(J)**2
    XI14S= 0.25*CS(J) + XIJ(J)
    CMULT= 2.0*FF(J)*CG(J)
    SMULT= 2.0*FF(J)*SG(J)
    GUCJ = 1.0
    IF   (ABS(YS(J)).LE.0.0001 . AND .ABS(CG (J)).LE.0.0001)  GUCJ=0.5
    CXT   = CH2*CM(J) - CS(J)*CN(J)*(XI14S-XM)
```

```
      CZT   = CZT + CS(J)*CN(J)*CMULT  * GUCJ
      CYT   = CYT - CS(J)*CN(J)*SMULT  * GUCJ
      CMT   = CMT + CXT * CMULT        * GUCJ
      CNT   = CNT - CXT * SMULT        * GUCJ
      CLT   = CLT+CS(J)*CN(J)*(YS(J)*CG (J)+ZS(J)*SG (J))*2.0*EE(J)*GUCJ
  320 CONTINUE
      WRITE (NOUT,100) KK
      CZT   = CZT/REFA
      CZTS = SYMA * CZT
      CYT   = CYT/REFA
      CYTS = SYMB * CYT
      CMT   = CMT/(REFA*REFC)
      CMTS = SYMA * CMT
      CNT   = CNT/(REFA*REFC)
      CNTS = SYMB * CNT
      CLTS = SYMB * CLT / (2.0*REFA*REFS)
      WRITE  (NOUT,110)  CZTS, CYTS, CMTS, CNTS, CLTS
      IF (NTZS . EQ . 0)  GO TO  340
      L1    = 0
      L2    = 0
      J1    = 1
      J2    = NBZ
      DO  330  J=J1,J2
      GLCB = 1.0
      IF   (ABS(YB( J)).LE.0.0001)  GLCB = 0.5
      L1    = L2+1
      L2    = L2+NSBEA(J)
      CZT   = CZT + CZB(J) * GLCB
      CMT   = CMT + (CMB(J)-CZB(J)*(XIS1(L1)-XM)/REFC) * GLCB
      CLT   = CLT + REFA * CZB(J) * YB(J) * GLCB
  330 CONTINUE
  340 CONTINUE
      IF (NTYS . EQ . 0)  GO TO  380
      L1    = 0
      L2    = 0
      J1    = NB-NBY+1
      J2    = NB
      J1M1  = J1-1
      IF  (J1M1.EQ.0)  GO TO  360
      DO  350  JX=1,J1M1
      L2    = L2+NSBEA(JX)
  350 CONTINUE
  360 CONTINUE
      DO  370  J=J1,J2
      GLCB = 1.0
      IF   (ABS(YB( J)).LE.0.0001)  GLCB = 0.5
      L1    = L2+1
      L2    = L2+NSBEA(J)
      CYT   = CYT + CYB(J) * GLCB
      CNT   = CNT + (CNB(J)-CYB(J)*(XIS1(L1)-XM)/REFC) * GLCB
      CLT   = CLT - REFA * CYB(J) * ZB(J) * GLCB
  370 CONTINUE
```

```
380 CONTINUE                                                          AERO 258
    CZT  = SYMA * CZT                                                 AERO 259
    CYT  = SYMB * CYT                                                 AERO 260
    CMT  = SYMA * CMT                                                 AERO 261
    CNT  = SYMB * CNT                                                 AERO 262
    CLT  = SYMB * CLT / (2.0*REFS*REFA)                               AERO 263
    WRITE  (NOUT,120)                                                 AERO 264
    WRITE  (NOUT,110)  CZT , CYT , CMT , CNT , CLT                    AERO 265
390 CONTINUE                                                          AERO 266
    RETURN                                                            AERO 267
    END                                                               AERO 268
```

```
      SUBROUTINE ATAN3(Y,X,T)
C  ***   COMPUTES   T=ATAN(Y/X)
      PI = 3.141593
      PI2 = PI*2.0
      IF(X.EQ.0.)  T=PI/2.
      IF(X.EQ.0.)  GO TO 2
      IF(Y.EQ.0.)T=0.
      IF(Y.EQ.0.0) GO TO 2
      A = Y/X
      T = ATAN(A)
    2 CONTINUE
      IF((Y.GE.0.).AND.(X.EQ.0.))  GO TO 1
      IF  ((Y.EQ.0.).AND.(X.GT.0.))  T=0.
      IF  ((Y.EQ.0.).AND.(X.GT.0.))  GO TO 1
      IF((Y.GE.0.).AND.(X.LE.0.))T = PI + T
      IF((Y.GE.0.).AND.(X.LE.0.))GO TO 1
      IF((Y.LE.0.).AND.(X.LE.0.))T = PI + T
      IF((Y.LE.0.).AND.(X.LE.0.))GO TU 1
      IF((Y.LE.0.).AND.(X.GE.0.))T = PI2+ T
      IF((Y.LE.0.).AND.(X.GE.0.))GO TO 1
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE BFSMAT(LENGTH ,    NE     ,    NB     ,    NP     ,    NTP    ,BFSMAT 2
     1         NTOTAL,    IO     ,    IOA    ,    FMACH  ,    CBAR   ,    KR     ,BFSMAT 3
     2         YB, ZB,    YS     ,    ZS     ,    X      ,    DELX   ,    FE     ,BFSMAT 4
     3         XIC   ,    SG     ,    CG     ,    AR     ,    RIA    ,    AO     ,BFSMAT 5
     4         XIS1  ,    XIS2   ,    AVR    ,    NSARAY,     NCARAY,     NSBEA  ,BFSMAT 6
     5         NBEA  ,    NAS    ,    NASB   ,    BFS    ,    BFSA   ,   SCALER) BFSMAT 7
C-----------------------------------------------------------------------BFSMAT 8
C        LENGTH       NTOTAL + THE TOTAL NUMBER OF Z- AND Y-ORIENTED         BFSMAT 9
C                     SLENDER BODY ELEMENTS                                  BFSMAT10
C        NE           GROUND EFFECTS FLAG                                    BFSMAT11
C        NB           NUMBER OF BODIES                                       BFSMAT12
C        NP           NUMBER OF PANELS                                       BFSMAT13
C        NTP          NUMBER OF LIFTING SURFACE BOXES                        BFSMAT14
C        NTOTAL       NTP + TOTAL NO. OF Y AND Z ORIENTED BODY ELEMENTS      BFSMAT15
C        IO           TAPE NUMBER ON WHICH THE  BFS   MATRIX IS WRITTEN      BFSMAT16
C        IOA          TAPE NUMBER ON WHICH THE  BFSA  MATRIX IS WRITTEN      BFSMAT17
C        FMACH        MACH NUMBER                                            BFSMAT18
C        CBAR         REFERENCE CHORD                                        BFSMAT19
C        KR           REDUCED FREQUENCY                                      BFSMAT20
C        YB           ARRAY OF -Y- COORDINATES OF THE BODIES                 BFSMAT21
C        ZB           ARRAY OF -Z- COORDINATES OF THE BODIES                 BFSMAT22
C        YS           ARRAY OF -Y- COORDINATES OF STRIPS AND BODIES          BFSMAT23
C        ZS           ARRAY OF -Z- COORDINATES OF STRIPS AND BODIES          BFSMAT24
C        X            ARRAY OF 3/4 CHORD LOCATIONS OF BOXES AND              BFSMAT25
C                     1/2 CHORD FOR BODY ELEMENTS                            BFSMAT26
C        DELX         ARRAY OF LENGTHS OF BOXES AND BODY ELEMENTS            BFSMAT27
C        EE           ARRAY OF THE SEMI-WITH OF STRIPS                       BFSMAT28
C        XIC          ARRAY OF 1/4 CHORD COORDINATES OF BOXES                BFSMAT29
C        SG           ARRAY OF SINE OF STRIP DIHEDRAL ANGLE                  BFSMAT30
C        CG           ARRAY OF COSINE OF STRIP DIHEDRAL ANGLE                BFSMAT31
C        AR           ARRAY OF RATIO OF MAJOR AXES OF BODIES                 BFSMAT32
C        RIA          ARRAY OF RADII OF BODY ELEMENTS                        BFSMAT33
C        AO           ARRAY OF SLENDER BODY ELEMENT RADII                    BFSMAT34
C        XIS1         ARRAY OF SLENDER BODY ELEMENT LEADING  EDGE COORD.S BFSMAT35
C        XIS2         ARRAY OF SLENDER BODY ELEMENT TRAILING EDGE COORD.S BFSMAT36
C         AVR         ARRAY OF RADII OF BODIES                               BFSMAT37
C        NSARAY       ARRAY OF THE NUMBER OF STRIPS PER PANEL                BFSMAT38
C        NCARAY       ARRAY OF THE NUMBER OF CHORDWISE DIV. PER PANEL        BFSMAT39
C        NSBEA        ARRAY OF THE NUMBER OF ELEMENTS PER SLENDER BODY       BFSMAT40
C        NBEA1        ARRAY OF NUMBER OF BODY ELEMENTS PER BODY              BFSMAT41
C        NBEA2        ARRAY OF THE BODY ORIENTATION FLAGS PER BODY           BFSMAT42
C        BFS          WORK ARRAY FOR TEMPORARY STORAGE OF THE BFS COLS.      BFSMAT43
C        BFSA         WORK ARRAY FOR TEMPORARY STORAGE OF THE BFSA COLS.     BFSMAT44
C                                                                            BFSMAT45
C-----------------------------------------------------------------------BFSMAT46
      REAL         KR                                                        BFSMAT47
      DIMENSION    YB   ( 1 ), ZB   ( 1 ), YS   ( 1 ), ZS   ( 1 ), X    ( 1 ), BFSMAT48
     1             DELX( 1 ), EE   ( 1 ), XIC  ( 1 ), SG   ( 1 ), CG   ( 1 ), BFSMAT49
     2             AR   ( 1 ), RIA ( 1 ), AVR  ( 1 ), XIS1( 1 ), XIS2( 1 ), BFSMAT50
     3             AO   ( 1 ), NAS  ( 1 ), NASB( 1 ), NBEA( 2,1),             BFSMAT51
     4             NSBEA(1)  , NCARAY(1), NSARAY(1)                          BFSMAT52
      COMPLEX      BFS (LENGTH, 2), FWZ , FWY , EIKJ1, EIKJ2                 BFSMAT53
```

```
      COMPLEX     BFSA(LENGTH, 2), FWZA, FWYA                           BFSMAT54
      LOGICAL        LAST                                               BFSMAT55
C-----------------------------------------------------------------------BFSMAT56
      IFLAG      = 0                                                    BFSMAT57
      NUMBR      = 0                                                    BFSMAT58
      REWIND   IO                                                       BFSMAT59
      REWIND   IOA                                                      BFSMAT60
      NPOT       = 6                                                    BFSMAT61
      BETA2      =      1.0   -   FMACH**2                              BFSMAT62
      IF1        =      2                                               BFSMAT63
      ICOL       =      0                                               BFSMAT64
      NTY        =      NTOTAL / 2                                      BFSMAT65
      GO  TO  140                                                       BFSMAT66
C-----------------------------------------------------------------------BFSMAT67
C-----------------------------------------------------------------------BFSMAT68
   10 CONTINUE                                                          BFSMAT69
C                      - Y - ORIENTED BODIES AS SENDING ELEMENTS        BFSMAT70
      SGS        =     -1.0                                             BFSMAT71
      CGS        =      0.0                                             BFSMAT72
      NASD       =      0                                               BFSMAT73
      IZYFLG     =      3                                               BFSMAT74
      ASSIGN   190 TO   IBODY                                           BFSMAT75
      ASSIGN  100 TO    ICOLMN                                          BFSMAT76
      GO   TO   80                                                      BFSMAT77
   20 CONTINUE                                                          BFSMAT78
C                      - LIFTING SURF. BOXES AS SENDING ELEMENTS        BFSMAT79
C                                                                       BFSMAT80
      IF ( NTP  .LE.  0  )   GO  TO     70                              BFSMAT81
      ASSIGN   30 TO    ICOLMN                                          BFSMAT82
      J          =      1                                               BFSMAT83
      JP1        =      J                                               BFSMAT84
      IBOX       =      0                                               BFSMAT85
      ISTRIP     =      0                                               BFSMAT86
C                                        LOOP  FOR - PANEL -            BFSMAT87
      DO        60   ISP   =    1, NP                                   BFSMAT88
      NS         =      NSARAY (ISP)                                    BFSMAT89
      NC         =      NCARAY (ISP)                                    BFSMAT90
      NASD       =      NAS    (ISP)                                    BFSMAT91
C                                        LOOP  FOR - STRIP -            BFSMAT92
      DO        50   IS    =    1, NS                                   BFSMAT93
      ISTRIP     =      ISTRIP   +  1                                   BFSMAT94
      DYS        =      YS ( ISTRIP )                                   BFSMAT95
      DZS        =      ZS ( ISTRIP )                                   BFSMAT96
      SGS        =      SG ( ISTRIP )                                   BFSMAT97
      CGS        =      CG ( ISTRIP )                                   BFSMAT98
      WIDTH      =      2.0 *  EE (ISTRIP )                             BFSMAT99
C                                                                       BFSMA100
C                                        LOOP  FOR - BOX   -            BFSMA101
      DO        40   IB    =    1, NC                                   BFSMA102
      IBOX       =      IBOX   +  1                                     BFSMA103
      DXS        =      XIC ( IBOX )                                    BFSMA104
C                                                                       BFSMA105
```

```
      ICOL      =     ICOL   +  1                                              BFSMA106
      GO TO  180                                                               BFSMA107
   30 CONTINUE                                                                 BFSMA108
      AREA      =     WIDTH   *   DELX (IBOX)                                  BFSMA109
      BFS(ICOL, 1)   =       BFS(ICOL, 1)   *   AREA                           BFSMA110
      BFSA(ICOL,1)   =       BFSA(ICOL,1)   *   AREA                           BFSMA111
      BFS(ICOL, 2)   =       BFS(ICOL,   2) *   AREA                           BFSMA112
      BFSA(ICOL,2)   =       BFSA(ICOL, 2)  *   AREA                           BFSMA113
   40 CONTINUE                                                                 BFSMA114
   50 CONTINUE                                                                 BFSMA115
   60 CONTINUE                                                                 BFSMA116
C                 - Z -  ORIENTEC BODIES AS SENDING ELEMENTS                   BFSMA117
C                                                                              BFSMA118
   70 CONTINUE                                                                 BFSMA119
      SGS       =     0.0                                                      BFSMA120
      CGS       =     1.0                                                      BFSMA121
      NASD      =     0                                                        BFSMA122
      IZYFLG    =     1                                                        BFSMA123
      ASSIGN   10 TO   IBODY                                                   BFSMA124
      ASSIGN  100 TO    ICOLMN                                                 BFSMA125
      GO  TO   80                                                              BFSMA126
C------------------------------------------------------------------------------BFSMA127
C------------------------------------------------------------------------------BFSMA128
C                                                                              BFSMA129
C    ***   LOOP FOR EACH  INTERFERENCE BODY SENDING ELEMENT                    BFSMA130
C                                                                              BFSMA131
   80 CONTINUE                                                                 BFSMA132
      INDEX     =     NTP                                                      BFSMA133
C                            -- ISB --   IS THE SENDING BODY                   BFSMA134
      DO  130 ISB  =     1, NB                                                 BFSMA135
      IF  (NBEA(2,ISB).EQ.2) GO TO 90                                          BFSMA136
      IF  (NBEA(2,ISB).NE.IZYFLG) GO TO 120                                    BFSMA137
   90 DYS       =     YB     (ISB)                                             BFSMA138
      NSBE  = NBEA(1,ISB)                                                      BFSMA139
      JP1       =     1                                                        BFSMA140
      LAST      =     .FALSE.                                                  BFSMA141
      DZS       =     ZB     (ISB)                                             BFSMA142
C                            -- ISBE -- IS THE ELEMENT OF THE SEND. BODYBFSMA143
      EARG2     =   1.0                                                        BFSMA144
      DO  110 ISBE =    1, NSBE                                                BFSMA145
      EARG1     =   EARG2                                                      BFSMA146
      INDEX     =     INDEX   +  1                                             BFSMA147
      DXS       =     X (INDEX) - DELX(INDEX) /4.0                             BFSMA148
      EARG2     =   KR * DELX(INDEX) / CBAR                                    BFSMA149
C                     CALCULATE THIS COLUMN                                    BFSMA150
      ICOL      =     ICOL   +  1                                              BFSMA151
      BFS(ICOL,1) = 0.0                                                        BFSMA152
      BFSA(ICOL,1)= 0.0                                                        BFSMA153
      BFS(ICOL,2) = 0.0                                                        BFSMA154
      BFSA(ICOL,2)= 0.0                                                        BFSMA155
      FIKJ1     =   CMPLX ( COS(EARG1), -SIN(EARG1) )                          BFSMA156
      FIKJ2     =   CMPLX ( COS(EARG2),  SIN(EARG2) )                          BFSMA157
```

```
      GO  TO  180                                                       BFSMA158
  100 CONTINUE                                                          BFSMA159
C                   IS THIS THE FIRST COLUMN, YES   BRANCH              BFSMA160
      IF ( ISBE  .EQ.   1 )   GO TO 110                                 BFSMA161
      BFS(ICOL-1, 1) = BFS(ICOL-1, 1)*EIKJ1 - BFS( ICOL,  1)*EIKJ2      BFSMA162
      BFSA(ICOL-1,1) = BFSA(ICOL-1,1)*EIKJ1 - BFSA( ICOL,1)*EIKJ2       BFSMA163
      BFS(ICOL-1, 2) = BFS(ICOL-1, 2)*EIKJ1 - BFS( ICOL,  2)*EIKJ2      BFSMA164
      BFSA(ICOL-1,2) = BFSA(ICOL-1,2)*EIKJ1 - BFSA( ICOL,2)*EIKJ2       BFSMA165
  110 CONTINUE                                                          BFSMA166
      GO TO  130                                                        BFSMA167
  120 CONTINUE                                                          BFSMA168
      INDEX = INDEX + NBEA(1,ISB)                                       BFSMA169
  130 CONTINUE                                                          BFSMA170
C                RETURN TO CALLING POINT - EITHER Y OR Z  SENDING BODY ELEMBFSMA171
C                                                                       BFSMA172
C   ***    GO   EITHER TO THE  Y-ORIENTED INTERFERENCE BODY ELEMENT LOOP  BFSMA173
C          OR   TO THE LOOP FOR SLENDER BODY SENDING ELEMENTS           BFSMA174
C                                                                       BFSMA175
      GO TO  IBODY,    (10,190)                                         BFSMA176
C                        CALCULATE EACH ROW OF THE SENDING COLUMN       BFSMA177
C-----------------------------------------------------------------------BFSMA178
C-----------------------------------------------------------------------BFSMA179
  140 CONTINUE                                                          BFSMA180
      IY         =      0                                               BFSMA181
      IROW       =      0                                               BFSMA182
C                            -- IRB -- IS THE RECEIVING BODY            BFSMA183
      DO  170 IRB  =    1,   NB                                         BFSMA184
      NRBE       =      NSBEA(IRB)                                      BFSMA185
C                                                                       BFSMA186
      XYB        =    YB( IRB )                                         BFSMA187
      XZB        =    ZB( IRB )                                         BFSMA188
C                                                                       BFSMA189
C                            -- IRBE -- IS THE ELEM. OF THE REC. BODY   BFSMA190
      DO  160 IRBE =   1,   NRBE                                        BFSMA191
      IY         =      IY   +   1                                      BFSMA192
      IROW       =      IROW   +  2                                     BFSMA193
      DRIA       =      AO   ( IY )                                     BFSMA194
      DXLE       =      XIS1( IY )                                      BFSMA195
      DXTE       =      XIS2( IY )                                      BFSMA196
      XX1        =    DXLE                                              BFSMA197
      XX2        =    DXTE                                              BFSMA198
      XAA        =    DRIA                                              BFSMA199
      ICOL       =      0                                               BFSMA200
      GO   TO  20                                                       BFSMA201
  150 CONTINUE                                                          BFSMA202
C                                                                       BFSMA203
      CALL        RWREC( IFLAG,   IO   , BFS( 1,1), LENGTH  , 1, 0    )  BFSMA204
      CALL        RWREC( IFLAG,   IO   , BFS( 1,2), LENGTH  , 1, 0    )  BFSMA205
      CALL        RWREC( IFLAG,   IOA  , BFSA(1,1), LENGTH  , 1, 0    )  BFSMA206
      CALL        RWREC( IFLAG,   IOA  , BFSA(1,2), LENGTH  , 1, 0    )  BFSMA207
C                                                                       BFSMA208
  160 CONTINUE                                                          BFSMA209
```

```
  170 CONTINUE                                                          BFSMA210
      RETURN                                                            BFSMA211
  180 CONTINUE                                                          BFSMA212
      CALL          FWMW   (  ND,    NE,      SGS,    CGS ,     IRB,    DRIA    ,  BFSMA213
     1    AR      , DXLE     , DXTE     , YB       , ZB        , DXS       ,  BFSMA214
     2    DYS     , DZS      , NASD     , NASB     , KR        , BETA2     ,  BFSMA215
     3    CBAR    , AVR      , FWZ,FWZA, FWY,FWYA, IF1,NUMBR, SCALER)  BFSMA216
      BFS( ICOL   , 1)   =    FWZ * ( DXTE - DXLE )                     BFSMA217
      BFSA(ICOL   , 1)   =    FWZA* ( DXTE - DXLE )                     BFSMA218
      BFS( ICOL   , 2)   =    FWY * ( DXTE - DXLE ) * (AR(IRB)**2)      BFSMA219
      BFSA(ICOL   , 2)   =    FWYA* ( DXTE - DXLE ) * (AR(IRB)**2)      BFSMA220
      GO  TO   ICOLMN, (100,30)                                         BFSMA221
C-----------------------------------------------------------------------BFSMA222
  190 CONTINUE                                                          BFSMA223
C                                                                       BFSMA224
C  ***   LOOP FOR EACH  SLENDER      BODY SENDING ELEMENT               BFSMA225
C                                                                       BFSMA226
      IZYFLG    =  1                                                    BFSMA227
      SGS       =  0.0                                                  BFSMA228
      CGS       =  1.0                                                  BFSMA229
  200 CONTINUE                                                          BFSMA230
      LSBE      =  0                                                    BFSMA231
      DO   240  LSB = 1, NB                                             BFSMA232
C  ***   LSB  IS THE INDEX OF THE SLENDER SENDING BODY                  BFSMA233
      IF   (NSBEA(LSB) . EQ . 0)  GO TO  240                            BFSMA234
      IF  (NBEA(2,LSB).EQ.2) GO TO 210                                  BFSMA235
      IF  (NBEA(2,LSB).NE.IZYFLG) GO TO 230                             BFSMA236
  210 CONTINUE                                                          BFSMA237
      XETA      =  YB( LSB )                                            BFSMA238
      XZETA     =  ZB( LSB )                                            BFSMA239
      MSBE      =  NSBEA(LSB)                                           BFSMA240
      DO   220  LSBS = 1, MSBE                                          BFSMA241
      LSBE      =  LSBE + 1                                             BFSMA242
      ICOL      =  ICOL + 1                                             BFSMA243
      BFS(ICOL,1) = 0.0                                                 BFSMA244
      BFS(ICOL,2) = 0.0                                                 BFSMA245
      XXIJ      =  0.50 * XIS1(LSBE) + 0.50 * XIS2(LSBE)                BFSMA246
      CALL          FWMW   (  ND,    NE,      SGS,    CGS ,     IRB,    DRIA    ,  BFSMA247
     *    AR      , DXLE     , DXTE     , YB       , ZB        , XXIJ      ,  BFSMA248
     *    XETA    , XZETA    , NUMBR    , NASB     , KR        , BETA2     ,  BFSMA249
     *    CBAR    , AVR      , FWZ,FWZA, FWY,FWYA, IF1,NUMBR, SCALER)  BFSMA250
      BFS(ICOL, 1)  =  FWZ  *  ( DXTE - DXLE )                          BFSMA251
      BFSA(ICOL, 1) =  FWZA *  ( DXTE - DXLE )                          BFSMA252
      BFS(ICOL, 2)  =  FWY  *  ( DXTE - DXLE ) * (AR(IRB)**2)           BFSMA253
      BFSA(ICOL, 2) =  FWYA *  ( DXTE - DXLE ) * (AR(IRB)**2)           BFSMA254
  220 CONTINUE                                                          BFSMA255
      GO TO  240                                                        BFSMA256
  230 CONTINUE                                                          BFSMA257
      LSBE      =  LSBE + NSBEA(LSB)                                    BFSMA258
  240 CONTINUE                                                          BFSMA259
      IF   (IZYFLG . EQ .  3)  GO TO  250                               BFSMA260
      IZYFLG   = 3                                                      BFSMA261
```

```
      SGS        =  -1.0                                                BFSMA262
      CGS        =   0.0                                                BFSMA263
      GO TO  200                                                        BFSMA264
  250 CONTINUE                                                          BFSMA265
C                                                                       BFSMA266
      GO TO  150                                                        BFSMA267
C                                                                       BFSMA268
      END                                                               BFSMA269
```

```
      SUBROUTINE CTLS(NEAASS, NDAS  , ND, ELXI, ELYI, ELZI, XA, YA, ZA,
     1               XB, YB, ZB, XAR, YAR, ZAR, XBR, YBR, ZBR,
     2               XI, ETA   ,                               NTP6       )
C
C        THIS SUBROUTINE COMPUTES THE XI- AND ETA-COORDINATES OF A
C        *GENERATED NODE* ON ELASTIC AXIS OF A CONTROL SURFACE (FOR
C        WHICH NO INPUT NODE EXISTS) FROM THE ASSOCIATED NODE (NODASS)
C        OF THE ASSOCIATED ELASTIC AXIS  (NEAASS)
C
      DIMENSION ELXI(1), ELYI(1), ELZI(1)
      DIMENSION  XAR( 1), YAR( 1), ZAR( 1), XBR( 1), YBR( 1), ZBR( 1)
     1        ,  XI( 1), ETA( 1)
C
      AXA        =  XAR(NEAASS)
      AYA        =  YAR(NEAASS)
      AZA        =  ZAR(NEAASS)
      AXB        =  XBR(NEAASS)
      AYB        =  YBR(NEAASS)
      AZB        =  ZBR(NEAASS)
C
      AXI        =  ELXI(NDAS)
      AETA       =  ELYI(NDAS)
      AZET       =  ELZI(NDAS)
C
      YDIF       =  YB - YA
      ZDIF       =  ZB - ZA
      RHO        =  SQRT(YDIF**2 + ZDIF**2)
      COSG       =  YDIF / RHO
      SING       =  ZDIF / RHO
      YAB        =  0.0
      YBB        =  YDIF*COSG + ZDIF*SING
      AYAW       =  (AYA - YA)*COSG + (AZA - ZA)*SING
      AYBW       =  (AYB - YA)*COSG + (AZB - ZA)*SING
      AETAW      =  (AETA- YA)*COSG + (AZET- ZA)*SING
C
      AMA        =  (AXB - AXI) / (AYBW - AETAW)
      AMAP       =  (AXB - AXA) / (AYBW - AYAW )
      TEST1      =  ABS(AMA - AMAP)
      IF (TEST1 .GE. 0.01) GO TO 10
C
      TEST2      =  ABS(AXI - AMA*AETAW - AXA + AMAP*AYAW)
      IF (TEST2 .LT. 0.01) GO TO 30
   10 CONTINUE
      WRITE  (NTP6,20)
   20 FORMAT  ( /// 10X, 70H***   WARNING    ***  NODAL POINT(S) DO NOT L
     1IE EXACTLY ON ELASTIC AXIS    //   )
C
   30 CONTINUE
      AYDIF      =  AYB - AYA
      AZDIF      =  AZB - AZA
      ARHO       =  SQRT(AYDIF**2 + AZDIF**2)
      ACOSG      =  AYDIF / ARHO
```

```
      ASING     =   AZDIF / ARHO                                              CTLS  54
      TEST3     =   ABS(ASING - SING)                                         CTLS  55
      TEST4     =   ABS(ACOSG - COSG)                                         CTLS  56
      IF  (TEST3 .GT. 0.001 .OR. TEST4 .GT. 0.001)  WRITE  (NTP6,70)          CTLS  57
C                                                                             CTLS  58
      AETYDF    =   AETAW - YAB                                               CTLS  59
      AM        =   (XB - XA) / (YBB - YAB)                                   CTLS  60
      DENOM     =   1.0 + AM*AMA                                              CTLS  61
C                                                                             CTLS  62
      XI(ND)    =   (AM*AETYDF + XA + AM*AMA*AXI) / DENOM                     CTLS  63
      ETA(ND)   =   (AMA*(AXI - XA) + AETAW + AM*AMA*YAB) / DENOM             CTLS  64
C                                                                             CTLS  65
      WRITE  (NTP6,40)  ND, NEAASS, NDAS  , XA , YAB, XB , YBB                CTLS  66
      WRITE  (NTP6,50)  AXA, AYA, AZA, AXB, AYB, AZB, AXI, AETA, AZET         CTLS  67
      WRITE  (NTP6,60)  AMA, AMAP, AM, AETAW, AETYDF, XI(ND), ETA(ND)         CTLS  68
C                                                                             CTLS  69
   40 FORMAT  ( // 4X, 38HND, NEAASS, NDAS  , XA , YAB, XB , YBB  /           CTLS  70
     1        3I6, 4E16.6 / )                                                 CTLS  71
   50 FORMAT  ( /  6X, 28HAXA, AYA, AZA, AXB, AYB, AZB  / 6F12.5  /           CTLS  72
     1          /  6X, 28HAXI, AETA, AZET               / 6F12.5  /   )       CTLS  73
   60 FORMAT  ( /  6X, 28HAMA, AMAP, AM, AETAW, AETYDF  / 5F12.5  /           CTLS  74
     1          /  6X, 28HXI(ND), ETA(ND)               / 5F12.5  /   )       CTLS  75
   70 FORMAT  ( /// 10X, 78H***  WARNING   ***  THE TWO ELASTIC AXES DOCTLS  76
     1 NOT LIE EXACTLY IN THE SAME PLANE     //   )                           CTLS  77
      RETURN                                                                  CTLS  78
      END                                                                     CTLS  79
```

```
      SUBROUTINE DOTP (NTPH,NTPDH,NTPH4,MASTH,MASTDH,MASTH4,IFNEWH,       DOTP   2
     1 NSYM,NASYM,NODES,MODES,NB,IROW,                                    DOTP   3
     2 PHIN,PHIZ,PHIY,COL,WORK )                                          DOTP   4
C                                                                         DOTP   5
      DIMENSION  PHIN(NODES, MODES), PHIZ(NODES, MODES),                  DOTP   6
     1 PHIY(NODES,MODES), COL(IROW), WORK(1)                              DOTP   7
C                                                                         DOTP   8
      REWIND  NTPH                                                        DOTP   9
      REWIND  NTPDH                                                       DOTP  10
C                                                                         DOTP  11
      IF (NB.EQ.0) GO TO 70                                               DOTP  12
      MASTAP     =  MASTH                                                 DOTP  13
      NTAPE     =  NTPH                                                   DOTP  14
      IGO       =  1                                                      DOTP  15
   10 REWIND  NTAPE                                                       DOTP  16
C                                                                         DOTP  17
      CALL ORGN (NTAPE,MASTAP,NSYM,NASYM,NODES,MODES, IROW,               DOTP  18
     1 COL,PHIZ,WORK )                                                    DOTP  19
C                                                                         DOTP  20
      GO TO  110                                                          DOTP  21
C                                                                         DOTP  22
   30 CONTINUE                                                            DOTP  23
      MASTAP    =  MASTDH                                                 DOTP  24
      NTAPE     =  NTPDH                                                  DOTP  25
      IGO       =  2                                                      DOTP  26
      GO TO  10                                                           DOTP  27
C                                                                         DOTP  28
   40 CONTINUE                                                            DOTP  29
      MASTAP     =  MASTH                                                 DOTP  30
      NTAPE     =  NTPH                                                   DOTP  31
      IGO       =  3                                                      DOTP  32
   50 REWIND  NTAPE                                                       DOTP  33
C                                                                         DOTP  34
      CALL ORGN (NTAPE,MASTAP,NSYM,NASYM,NODES,MODES, IROW,               DOTP  35
     1 COL,PHIY,WORK )                                                    DOTP  36
C                                                                         DOTP  37
      GO TO  110                                                          DOTP  38
C                                                                         DOTP  39
   60 CONTINUE                                                            DOTP  40
      MASTAP    =  MASTDH                                                 DOTP  41
      NTAPE     =  NTPDH                                                  DOTP  42
      IGO       =  4                                                      DOTP  43
      GO TO  50                                                           DOTP  44
C                                                                         DOTP  45
   70 CONTINUE                                                            DOTP  46
      IF  (IFNEWH .NE. 0)   RETURN                                        DOTP  47
      MASTAP     =  MASTH                                                 DOTP  48
      NTAPE     =  NTPH                                                   DOTP  49
      IGO       =  5                                                      DOTP  50
   80 CONTINUE                                                            DOTP  51
C                                                                         DOTP  52
      CALL ORGN (NTAPE,MASTAP,NSYM,NASYM,NODES,MODES, IROW,               DOTP  53
```

```
     1 COL,PHIN,WORK )                                                  DOTP  54
C                                                                       DOTP  55
      GO TO  110                                                        DOTP  56
   90 CONTINUE                                                          DOTP  57
      MASTAP    =   MASTDH                                              DOTP  58
      NTAPE     =   NTPDH                                               DOTP  59
      IGO       =   6                                                   DOTP  60
      GO TO  80                                                         DOTP  61
  100 CONTINUE                                                          DOTP  62
      MASTAP    =   MASTH4                                              DOTP  63
      NTAPE     =   NTPH4                                               DOTP  64
      REWIND  NTAPE                                                     DOTP  65
      IGO       =   7                                                   DOTP  66
      GO TO  80                                                         DOTP  67
  110 CONTINUE                                                          DOTP  68
C                                                                       DOTP  69
C                                                                       DOTP  70
      GO TO  (30, 40, 60, 70, 90, 100, 120),   IGO                      DOTP  71
  120 CONTINUE                                                          DOTP  72
      REWIND  MASTH                                                     DOTP  73
      REWIND  MASTDH                                                    DOTP  74
      REWIND  MASTH4                                                    DOTP  75
      RETURN                                                            DOTP  76
      END                                                               DOTP  77
```

```
      SUBROUTINE DPPS(K,KS,I,J1,J2,SGR,CGR,REFC,FMACH,YS,ZS,NBARAY,
     1   NCARAY,DT,DTA
     2 , YB,ZB,ARB,AVR,XLE,XTE,X,CG,EE,SG,XIC,DELX,XLAM,   SCALER )
      DIMENSION YB(1), ZB(1), ARB(1), AVR(1), XLE(1), XTE(1), X(1)
      DIMENSION CG(1), EE(1), SG(1), XIC(1), DELX(1)
      DIMENSION XLAM(1)
C   ***   GENERATES ROWS OF THE  DPP  SUBMATRIX USING
C         SUBROUTINE  SUBP
      REAL         KR
      DIMENSION    YS( 1 ), ZS( 1 ), NBARAY( 1 ), NCARAY( 1 )
      COMPLEX      SUM(2), DT( 1 ), DTA( 1 )
      BETA = SQRT(1.0-FMACH**2)
      FL   = REFC
      L    = 1
C   L IS THE PANEL NUMBER ASSOCIATED WITH SENDING   POINT  J
      LS   = 1
C   LS IS THE STRIP NUMBER ASSOCIATED WITH SENDING   POINT  J
      IC   = 0
      NBXS = NBARAY(L)
      NC1  = NCARAY(L)
      NCPNB= NC1
      NBCUM= NC1
      YREC = YS(KS)
      ZREC = ZS(KS)
      DO   20  J=J1,J2
      IC   = IC+1
      IR   = J
      CALL         SUBP (I,L,LS,J,IC,IR,NBXS,NCPNB,SGR,CGR,YREC,ZREC,SUM
     1 , NCARAY,YB,ZB,ARB,AVR,XLE,XTE,X,CG,EE,SG,YS,ZS,XIC,DELX,XLAM
     2 , SCALER                                                          )
      DT(J)= SUM(1)
      DTA(J)=SUM(2)
      IF  (J.EQ.J2)   GO TO   20
      IF  (IC.EQ.NC1)   IC=0
      IF  (J.LT.NBXS)    GO TO  10
      L     = L+1
      NC1   = NCARAY(L)
      NCPNB= NBXS + NC1
      NBXS  = NBARAY(L)
   10 CONTINUE
      IF  (J.LT.NBCUM)   GO TO  20
      LS    = LS+1
      NBCUM= NBCUM+NC1
   20 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DPZY(    KB,IZ,I,J1,J2,IFIRST,ILAST,REFC,FMACH,YB,ZB,    DPZY
     1               AVR,ARB,TH1A,TH2A,NT12,NBARAY,NCARAY,NZYKB,DPZ,DPY,      DPZY
     2   DPZA,DPYA                                                            DPZY
     3 , XLE, XTE, X, CG, EE, SG, YS, ZS, XIC, DELX, XLAM, SCALER )           DPZY
      DIMENSION XIC(1), DELX(1), XLAM(1)                                      DPZY
      DIMENSION XLE(1), XTE(1), X(1), CG(1), EE(1), SG(1), YS(1), ZS(1)       DPZY
C   ***   GENERATES ROWS OF THE SUBMATRICES  DPZ  AND DPY  USING              DPZY
C         SUBROUTINE  SUBP                                                    DPZY
      COMPLEX      SUM(2), DPZ( 1 ), DPY( 1 ), DPZA( 1 ), DPYA( 1 )           DPZY
      DIMENSION NT12(2,1), YB(1), ZB(1), AVR(1), ARB(1),                      DPZY
     *            TH1A( 1 ), TH2A( 1 ), NBARAY(1), NCARAY( 1 )                DPZY
   10 FORMAT   (1H0,15I5)                                                     DPZY
   20 FORMAT   (1H0,10X,6E18.8)                                               DPZY
      PI    = 3.1415926                                                       DPZY
      BETA  = SQRT(1.0-FMACH**2)                                              DPZY
      FL    = REFC                                                            DPZY
      NBV   = 0                                                               DPZY
      IX1     = 1                                                             DPZY
      IZ    = IZ+1                                                            DPZY
C   IZ  IS THE BODY-ELEMENT NUMBER FOR BODY  KB  --   IZ RUNS FROM  1         DPZY
C   THROUGH  NBE-SUB-KB                                                       DPZY
      JC    = 2                                                               DPZY
      IF   (IZ.GE.IFIRST .AND. IZ.LE.ILAST)  JC = 1                           DPZY
      IF   (KB .EQ. 1)  GO TO  28                                             DPZY
      KBM1 = KB-1                                                             DPZY
      DO  24  KX = 1, KBM1                                                    DPZY
   24 IX1  = IX1 + NT12(JC,KX)                                                DPZY
   28 CONTINUE                                                                DPZY
      IX2 = NT12(JC,KB) + IX1 - 1                                             DPZY
      DO  100  IX=IX1,IX2                                                     DPZY
      DPUR  = (0.0,0.0)                                                       DPZY
      DPUL  = (0.0,0.0)                                                       DPZY
      DPLR  = (0.0,0.0)                                                       DPZY
      DPLL  = (0.0,0.0)                                                       DPZY
      L     = 1                                                               DPZY
C   L IS THE PANEL NUMBER ASSOCIATED WITH SENDING   POINT  J                  DPZY
      LS    = 1                                                               DPZY
C   LS IS THE STRIP NUMBER ASSOCIATED WITH SENDING   POINT  J                 DPZY
      IC    = 0                                                               DPZY
      NBXS = NBARAY(L)                                                        DPZY
      NC1   = NCARAY(L)                                                       DPZY
      NCPNB= NC1                                                              DPZY
      NBCUM= NC1                                                              DPZY
      IXP1  = IX+1                                                            DPZY
      IF  (IXP1.GT.IX2)   IXP1= IX1                                           DPZY
      IXM1  = IX-1                                                            DPZY
      IF  (IX.EQ.IX1)     IXM1=IX2                                            DPZY
      IF  (IZ.GE.IFIRST.AND.IZ.LE.ILAST)  GO TO  30                           DPZY
      THETA= TH2A(IX)                                                         DPZY
      THP1  = TH2A(IXP1)                                                      DPZY
      THM1  = TH2A(IXM1)                                                      DPZY
      GO TO  40                                                               DPZY
```

```
30 CONTINUE                                                            DPZY  54
   THETA = THIA(IX)                                                    DPZY  55
   THP1  = THIA(IXP1)                                                  DPZY  56
   THM1  = THIA(IXM1)                                                  DPZY  57
40 CONTINUE                                                            DPZY  58
   IF  (IX.EQ.IX1)   THM1=THM1-2.0*PI                                  DPZY  59
   IF  (IX.EQ.IX2)   THP1=THP1+2.0*PI                                  DPZY  60
   DELTH= 0.5*(THP1  - THM1)                                           DPZY  61
   YREC  = YB(KB)+AVR(KB)*COS(THETA)                                   DPZY  62
   ZREC  = ZB(KB)+AVR(KB)*ARB(KB)*SIN(THETA)                           DPZY  63
   RHO   = SQRT(1.0+(ARB(KB)**2 - 1.0) * (COS(THETA))**2)              DPZY  64
   SCR   = -ARB(KB)*COS(THETA)/RHO                                     DPZY  65
   CCR   = SIN(THETA)/RHO                                              DPZY  66
   SMULT= SIN(THETA)  * RHO / PI                                       DPZY  67
   CMULT= COS(THETA)  * RHO / PI                                       DPZY  68
   DO   90   J=J1,J2                                                   DPZY  69
   IO    = IO+1                                                        DPZY  70
   IR    = J                                                           DPZY  71
   CALL        SUBP (I,L,LS,J,IO,IR,NBXS,NCPNB,SGR,CGR,YREC,ZREC,SUM   DPZY  72
  1 , NCARAY,YB,ZB,ARB,AVR,XLE,XTE,X,CG,FF,SG,YS,ZS,XIC,DELX,XLAM      DPZY  73
  2 , SCALER                                                         ) DPZY  74
   GO TO  (50,50,60),   NZYKB                                          DPZY  75
50 CONTINUE                                                            DPZY  76
   DPZ(J) = DPZ(J) + SUM(1) * SMULT * DELTH                            DPZY  77
   DPZA(J)= DPZA(J)+ SUM(2) * SMULT * DELTH                            DPZY  78
   IF  (NZYKB.EQ.1)   GO TO 70                                         DPZY  79
60 CONTINUE                                                            DPZY  80
   DPY(J) = DPY(J) + SUM(1) * CMULT * DELTH                            DPZY  81
   DPYA(J)= DPYA(J) +SUM(2) * CMULT * DELTH                            DPZY  82
70 CONTINUE                                                            DPZY  83
   IF  (J.EQ.J2)   GO TO  90                                           DPZY  84
   IF  (IO.EQ.NC1)   IO=0                                              DPZY  85
   IF  (J.LT.NBXS)    GO TO  80                                        DPZY  86
   L     = L+1                                                         DPZY  87
   NC1   = NCARAY(L)                                                   DPZY  88
   NCPNB= NBXS + NC1                                                   DPZY  89
   NBXS  = NBARAY(L)                                                   DPZY  90
80 CONTINUE                                                            DPZY  91
   IF  (J.LT.NBCUM)   GO TO  90                                        DPZY  92
   LS    = LS+1                                                        DPZY  93
   NBCUM= NBCUM+NC1                                                    DPZY  94
90 CONTINUE                                                            DPZY  95
100 CONTINUE                                                           DPZY  96
   RETURN                                                              DPZY  97
   END                                                                 DPZY  98
```

```
      SUBROUTINE  DUMULT   (    N1     ,    N2     ,   NTZS   ,   NTYS   ,    DUMULT 2
     *     W      ,    DZ      ,   UZ     ,    DY     ,   UY         )          DUMULT 3
CDUMULT          COMPLEX MATRIX MULTIPLICATION                                 DUMULT 4
C                W =  D7 * UZ  +  DY * UY                                      DUMULT 5
C                                                                              DUMULT 6
C                                                                              DUMULT 7
C  *** *    *    *    *    *    *    *    *    *    *    *    *    *    *     DUMULT 8
C                                                                              DUMULT 9
C         N1          FIRST ROW OF W TO CALCULATE                I*4           DUMULT10
C         N2          LAST ROW OF W TO CALCULATE                 I*4           DUMULT11
C         NTZS        NO. COL. IN DZ AND NO. ROWS IN UZ          I*4           DUMULT12
C         NTYS        NO. COL. IN DY AND NO. ROWS IN UY          I*4           DUMULT13
C         W           OUTPUT COLUMN                              COMPLEX       DUMULT14
C         DZ          DZ MATRIX  (ROWS N1 THRU N2  )               ...         DUMULT15
C         UZ          UZ MATRIX                                    ...         DUMULT16
C         DY          DY MATRIX  (ROWS N1 THRU N2  )               ...         DUMULT17
C         UY          UY MATRIX                                  COMPLEX       DUMULT18
C                                                                              DUMULT19
C  *** *    *    *    *    *    *    *    *    *    *    *    *    *    *     DUMULT20
C                                                                              DUMULT21
      COMPLEX        W(1)     , DZ (NTZS, 1 ),      UZ ( NTZS ) ,              DUMULT22
     *                          DY (NTYS, 1 ),      UY ( NTYS )                DUMULT23
C                                                                              DUMULT24
      IF ( NTZS .LE. 0 ) GO  TO   100                                          DUMULT25
      K        =     0                                                         DUMULT26
      DO      20     I  =  N1,  N2                                             DUMULT27
      K        =     K  +  1                                                   DUMULT28
      DO      10     J  =  1,  NTZS                                            DUMULT29
      W(I)     =   W(I)   +   DZ( J, K) * UZ( J )                              DUMULT30
   10 CONTINUE                                                                 DUMULT31
   20 CONTINUE                                                                 DUMULT32
C                                                                              DUMULT33
C                                                                              DUMULT34
  100 IF ( NTYS .LE. 0 )  GO  TO   200                                         DUMULT35
C                                                                              DUMULT36
C                W  + DY * UY                                                  DUMULT37
      K        =     0                                                         DUMULT38
      DO     120     I  =  N1,  N2                                             DUMULT39
      K        =     K + 1                                                     DUMULT40
      DO     110     J  =  1,  NTYS                                            DUMULT41
      W(I)     =   W(I)   +   DY( J, K ) * UY( J )                             DUMULT42
  110 CONTINUE                                                                 DUMULT43
  120 CONTINUE                                                                 DUMULT44
C                                                                              DUMULT45
  200 RETURN                                                                   DUMULT46
      END                                                                      DUMULT47
```

```
      SUBROUTINE DYPZ(KB,KS,LS,IZ,I,J1,J2,NYFLAG,FLND,FLNE,SGR,CGR,REFC,
     1    FMACH,KR,ARB,NBEA,LBO,LSO,JBO,DT,DTA
     2 , YB, ZB, RIA, X, YS, ZS, DELX )
      DIMENSION YB(1), ZB(1), RIA(1), X(1), YS(1), ZS(1), DELX(1)
C  ***    GENERATES ROWS OF THE SUBMATRICES  DYP, DYZ  AND DYY
C         USING  SUBROUTINE  SUBB
      REAL       KR,M
      COMPLEX    SUM(2), DT( 1 ), DTA( 1 )
      DIMENSION NBEA(2,1), ARB(1)
   10 FORMAT  (1H0,6E20.8)
      NDY   = 1
      NYFL  = NYFLAG
      IND   = 0
      PI    = 3.1415926
      EPS   = 0.00001
      BETA  = SQRT(1.0-FMACH**2)
      FL    = REFC
      M     = FMACH
      NBV   = 0
      JZ    = 0
      LB    = LBO
C   LB  IS THE BODY NUMBER ASSOCIATED WITH SENDING POINT  J
      SGS   = -1.0
      CGS   = 0.0
      LS    = LSO
C   LS IS THE INDEX OF THE  Y  AND  Z  COORDINATES OF SENDING POINT  J --
C   LS RUNS FROM NSTRIP+NB-NBY+1  THROUGH  NSTRIP+NB
      JB    = JBO-1
      NZYLB = NBEA(2,LB)
      AR    = ARB(LB)
      SL    = 0.0
      CL    = 1.0
      TL    = 0.0
      DO  20  J=J1,J2
      JB    = JB+1
      JZ    = JZ+1
      CALL         SUBB(KB,KS,I,J,JZ,JB,LB,LS,NDY,NYFL,FLND,FLNE,PI,EPS,
     1      SGR,CGR,SGS,CGS,AR,SL,CL,TL,FL,BETA,SUM
     2 , YB, ZB, RIA, X, YS, ZS, DELX )
      DT(J)= SUM(1)
      DTA(J)=SUM(2)
      IF  (JZ.EQ.NBEA(1,LB)) GO TO 20
      GO TO 30
   20 CONTINUE
      JZ    = 0
      LB    = LB+1
      LS    = LS+1
      AR    = ARB(LB)
      NZYLB = NBEA(2,LB)
   30 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DZPY(KB,KS,LS,IZ,I,J1,J2,NYFLAG,FLND,FLNE,SGR,CGR,REFC,
     1   FMACH,KR,ARB,NBEA,DT,DTA
     2 , YB, ZB, RIA, X, YS, ZS, DELX )
      DIMENSION YB(1), ZB(1), RIA(1), X(1), YS(1), ZS(1), DELX(1)
C  ***    GENERATES ROWS OF THE SUBMATRICES  DZP, DZZ  AND  DZY
C         USING  SUBROUTINE  SUBB
      REAL         KR,M
      COMPLEX      SUM(2), DT( 1 ), DTA( 1 )
      DIMENSION NBEA(2,1), ARB(1)
   10 FORMAT   (1H0,6F20.8)
      NDY    = 0
      NYFL   = NYFLAG
      IND    = 0
      NZYKB  = NBEA(2,KB)
      PI     = 3.1415926
      EPS    = 0.00001
      BETA   = SQRT(1.0-FMACH**2)
      FL     = REFC
      M      = FMACH
      NBV    = 0
      JZ     = 0
      LB     = 1
      JB     = 0
      SGS    = 0.0
      CGS    = 1.0
      NZYLB  = NBEA(2,LB)
      AR     = ARB(LB)
      SL     = 0.0
      CL     = 1.0
      TL     = 0.0
C    LS IS THE INDEX OF THE   Y  AND   Z  COORDINATES OF SENDING POINT  J --
C    LS RUNS FROM NSTRIP+1  THROUGH  NSTRIP+NBZ
      DO  30  J=J1,J2
      JB     = JB+1
C    JB  IS THE BODY-ELEMENT NUMBER IN BODY  LB  --  JB  RUNS FROM  1
C    THROUGH  NTZ
      JZ     = JZ+1
C    JZ  RUNS FROM 1  THROUGH  NBE-SUB-LB
      CALL          SUBB(KB,KS,I,J,JZ,JB,LB,LS,NDY,NYFL,FLND,FLNE,PI,EPS,
     1      SGR,CGR,SGS,CGS,AR,SL,CL,TL,FL,BETA,SUM
     2 , YB, ZB, RIA, X, YS, ZS, DELX )
      DT(J)   = SUM(1)
      DTA(J)  = SUM(2)
      IF  (JZ.EQ.NBEA(1,LB)) GO TO 20
      GO  TO 30
   20 CONTINUE
      JZ     = 0
      LB     = LB+1
      LS     = LS+1
      AR     = ARB(LB)
      NZYLB  = NBEA(2,LB)
   30 CONTINUE
```

```
      RETURN                                                           DZPY  54
      END                                                              DZPY  55
```

```
      SUBROUTINE     DZYMAT  ( D, DA ,   NFB    ,   NLB    ,   NTZYS ,        DZYMAT 2
     1   IDZDY, NTAPE, ITAP2, XP, BETA, IPRNT                                  DZYMAT 3
     2 , NBEA,       NSBE ,NC,NS ,AO,YB,ZB,AR,XIS1,XIS2,CG,SG,YP,ZP )          DZYMAT 4
      DIMENSION NBEA(2 ,1),                  NSBE(1), NC(1), NS(1), AO(1)      DZYMAT 5
      DIMENSION YB(1), ZB(1), AR(1), XIS1(1), XIS2(1), CG(1), SG(1)            DZYMAT 6
      DIMENSION YP(1), ZP(1)                                                   DZYMAT 7
CDZYMAT          CALCULATION OF DZ AND DY MATRICES                             DZYMAT 8
C                SLENDER BODY CALCULATIONS                                     DZYMAT 9
C *                                                                       *    DZYMAT10
C *      ********* **********************************************************  DZYMAT11
C *                                                                       *    DZYMAT12
C *      D           WORKING ARRAY USED TO STORE A ROW OF DZ OR DY        *    DZYMAT13
C *      NFB         NUMBER OF THE FIRST BODY WITH THE ORIENTATION REQ*        DZYMAT14
C *                  UESTED                                               *    DZYMAT15
C *      NLB         NUMBER OF THE LAST BODY WITH THE ORIENTATION         *    DZYMAT16
C *                  REQUESTED                                            *    DZYMAT17
C *      NTZYS       NUMBER OF Z OR Y ORIENTED SLENDER BODY ELE.          *    DZYMAT18
C *      NTAPE       I/O UNIT NUMBER WHICH THE OUTPUT MATRIX IS TO        *    DZYMAT19
C *                  BE WRITTEN ON                                        *    DZYMAT20
C *      XP          X-CONTROL POINT COORDINATE OF LIFTING SURFACE        *    DZYMAT21
C *                  BOXES                                                *    DZYMAT22
C *      BETA        SQRT( 1.0 - M**2)                                    *    DZYMAT23
C *                                                                       *    DZYMAT24
C *      ********* **********************************************************  DZYMAT25
C *                                                                       *    DZYMAT26
C ***    *    *    *    *    *    *    *    *    *    *    *    *    *    *    DZYMAT27
C ***    *    *    *    *    *    *    *    *    *    *    *    *    *    *    DZYMAT28
C                                                                              DZYMAT29
      COMMON /AROCOM/ NTI, MODES                                               DZYMAT30
     X , NP, MSTRIP, NSMAX, NCMAX, NTOTAL, NB, MSBE, MBE                       AROCOM 2
     Y , ND, NF, NBY, NBZ, NTO, NTP, NTY, NTZ                                  AROCOM 3
     1 , NTYS, NTZS, MAXGR, MAXSTR, NSBETO, NSTRIP, KR, XM, REFA, REFC         AROCOM 4
     2 , REFS, FMACH, LINES                                                    AROCOM 5
C                                                                              AROCOM 6
C                                                                              DZYMAT32
      DIMENSION      D(1), DA(1),XP(1)                                         DZYMAT33
      REAL         KP                                                          DZYMAT34
      INTEGER BY,   BZ,  C,   C1,  P,   S,   S1,  YT,  ZT                      DZYMAT35
C                                                                              DZYMAT36
      NPOT      =     6                                                        DZYMAT37
      C1       =    0                                                          DZYMAT38
      S1       =    0                                                          DZYMAT39
      BZ        =  0                                                           DZYMAT40
      BY        =  0                                                           DZYMAT41
      NFYB      =    NB - NBY  + 1                                             DZYMAT42
C                                                                              DZYMAT43
C                         THIS LOOP IS FOR EACH LIFTING SURF. PANEL            DZYMAT44
C                                                                              DZYMAT45
      DO     400     P  = 1,  NP                                               DZYMAT46
      NSP       =    NS(P)                                                     DZYMAT47
      NCP       =    NC(P)                                                     DZYMAT48
                                                                               DZYMAT49
```

```
C                                LOOP FOR EACH STRIP IN PANEL - P -        DZYMAT50
C                                                                          DZYMAT51
      DO     350     S  = 1,  NSP                                          DZYMAT52
      S1      =      S1 + 1                                                DZYMAT53
C                      Y AND Z COORDINATE OF STRIP                         DZYMAT54
      DY      =      YP(S1)                                                DZYMAT55
      DZ      =      ZP(S1)                                                DZYMAT56
      SGR     =       SG(S1)                                               DZYMAT57
      CGR     =       CG(S1)                                               DZYMAT58
C                                LOOP FOR EACH CHORDWISE ELEMENT IN STRIP  DZYMAT59
C                                                                          DZYMAT60
      DO     300     C  = 1,  NCP                                          DZYMAT61
      C1      =      C1  +  1                                              DZYMAT62
      DX      =      XP (C1)                                               DZYMAT63
C                                                                          DZYMAT64
C                              - ROWDYC -  CALCULATES ROW -C1- OF DZ OR DY DZYMAT65
C                                                                          DZYMAT66
      CALL      ROWDYZ  ( NFB    ,   NLB    ,   C1     ,   NTZYS ,         DZYMAT67
     *                   D, DA ,    DX     ,   DY     ,   DZ     ,   BZ   ,DZYMAT68
     2   BETA, IDZDY, NTAPE, NTAP2, SGR, CGR, IPRNT                        DZYMAT69
     3 ,        NSBE, AO, YB, ZB, AR, XIS1, XIS2 )                         DZYMAT70
C                                                                          DZYMAT71
  300 CONTINUE                                                             DZYMAT72
C                                                                          DZYMAT73
  350 CONTINUE                                                             DZYMAT74
C                                                                          DZYMAT75
  400 CONTINUE                                                             DZYMAT76
C                  WE HAVE NOW CALCULATED -C1- ROWS WHICH ARE THE          DZYMAT77
C                  LIFTING SURFACES.                                       DZYMAT78
C             NOW....LOOP FOR THE -Z- ORIENTED BODIES                      DZYMAT79
      IF ( NBZ  .LE.  0  )     GO  TO  510                                 DZYMAT80
      IF   ( NTZ . LE . 0 )    GO TO   510                                 DZYMAT81
      SGR     =      0.0                                                   DZYMAT82
      CGR     =      1.0                                                   DZYMAT83
      DO     500     BZ  =  1, NBZ                                         DZYMAT84
      DY      =     YB (BZ )                                               DZYMAT85
      DZ      =     ZB (BZ )                                               DZYMAT86
      NBEZ = NBEA(1,BZ)                                                    DZYMAT87
C                                LOOP FOR EACH ELEMENT OF BODY - BZ -      DZYMAT88
C                                                                          DZYMAT89
      DO     450     ZT  =  1,  NBEZ                                       DZYMAT90
      C1      =      C1 + 1                                                DZYMAT91
      DX       =      XP(  C1 )                                            DZYMAT92
C                                                                          DZYMAT93
      CALL      ROWDYZ  ( NFB    ,   NLB    ,   C1     ,   NTZYS ,         DZYMAT94
     *                   D, DA ,    DX     ,   DY     ,   DZ     ,   BZ   ,DZYMAT95
     2   BETA, IDZDY, NTAPE, NTAP2, SGR, CGR, IPRNT                        DZYMAT96
     3 ,        NSBE, AO, YB, ZB, AR, XIS1, XIS2 )                         DZYMAT97
  450 CONTINUE                                                             DZYMAT98
  500 CONTINUE                                                             DZYMAT99
C                                                                          DZYMA100
C               NOW..... LOOP FOR THE - Y - ORIENTED BODIES                DZYMA101
```

```
C
  510 IF ( NB .LT. NFYB )   GC TO  650
      IF   ( NTY . LE . 0 )   GO TO   650
      IXP       =     NTP
      IF (  NFYB  .LE.  1 )  GC TO 530
      NFYBM1    =    NFYB  -  1
      DC    520    I =    1,   NFYBM1
  520 IXP = IXP + NBEA(1,I)
  530 CONTINUE
      SGR       =    -1.0
      CGR       =     0.0
      DO    600   BY  =  NFYB, NB
      DY        =     YB (BY)
      DZ        =     ZB (BY)
      NBEY = NBEA(1,BY)
C
C                               LOOP FOR EACH ELEMENT OF BODY - BY -
      DO    550   YT   =   1,   NBEY
      C1        =     C1  +  1
      IXP       =      IXP  +  1
      DX        =     XP(  IXP  )
C
      CALL      ROWDYZ  ( NFB   ,   NLB   ,   C1   ,   NTZYS ,
     *                  D, DA ,    DX    ,   DY   ,   DZ    ,   BY    ,
     2   BETA, IDZDY, NTAPE, NTAP2, SGR, CGP,  IPRNT
     3 ,        NSBE, AO, YB, ZB, AR, XIS1, XIS2 )
C
  550 CONTINUE
  600 CONTINUE
  650 CONTINUE
C
      IF ( IPRNT  .NE.  0 )   WRITE ( NPOT, 1000)
     *    NP,     NBFZ,     NFYB,     NB,     IDZDY
      PETURN
C
 1000 FORMAT (16H DZYMAT***  NP =,I5, 11H NO. ROWS -,I5,15H FIRST Y BODY
     * - , I5,12H LAST BODY -, I5,  8H IDZDY =, I3 )
      END
```

```
      SUBROUTINE  DZY  ( X, Y, Z, SGR, CGR, XI1, XI2, ETA, ZETA, AR, AO,
     *   KR      , CBAR      , BETA      , FMACH     ,
     *    IDZDY,      DZDYR,      DZDYI,       LHS   ,       IPRNT    )
C
CDZY          CALCULATION OF THE DZ AND DY MATRICES
C             USED IN SLENDER BODY FLOW
C
C  *                                                                  *
C  *       *********  ************************************************
C  *                                                                  *
C  *       X           X- COORDINATE OF THE RECEIVING POINT           *
C  *       Y           Y - COORDINATE OF THE RECEIVING POINT          *
C  *       Z           Z - COORDINATE OF THE RECEIVING POINT          *
C  *       SGR         SINE OF THE RECEIVING POINT DIHEDRAL ANGLE     *
C  *       CGR         COSINE OF RECEIVING POINT DIHEDRAL ANGLE       *
C  *       XI1                                                        *
C  *       XI2                                                        *
C  *       ETA                                                        *
C  *       ZETA                                                       *
C  *       AR          ASPECT RATIO OF THE SENDING BODY               *
C  *       AO          RADIUS OF THE SENDING BODY                     *
C  *       KR          REDUCED FREQUENCY                              *
C  *       CBAR        REFERENCE CHORD LENGTH                         *
C  *       BETA        SQRT(1.0-M**2)                                 *
C  *       FMACH       MACH NUMBER                                    *
C  *       IDZDY       FLAG INDICATING WHETHER DZ OF DY IS TO BE      *
C  *                   CALCULATED.  ==0  DZ,     OTHERWISE DY         *
C  *       DZDYR       REAL PART OF DZ OR DY                          *
C  *       DZDYI       IMAGINARY PART OF DZ OR DY                     *
C  *                                                                  *
C  *       *********  ************************************************
C  *                                                                  *
      REAL    KD1PR, KD1PI, KD2PR, KD2PI, KD1MR, KD1MI, KD2MR, KD2MI, KR
C
      DATA   PI16 / 50.265482 /
C
C          THE COMPLEX NUMBERS IN THIS ROUTINE ARE TREATED SEPERATLY AS
C          THE REAL PART,  NAME APPENDED BY AN  -R- ,  AND THE
C          IMAGINARY PART, NAME APPENDED BY AN  -I- .
C
      DATA     NPOT   /6/
      IF   (  IPRNT   .NE.  0    )     WRITE (  NPOT , 100 )
     *     X, Y, Z, SGR,  CGR,    XI1,  XI2,  ETA,  ZETA ,
     *    AR, AO, KR, IDZDY
  100 FORMAT ( 73H DZDY INPUT-- X, Y, Z, SGR, CGR, XI1, XI2, ETA, ZETA,
     *AR, AO, KR, IDZDY        /  10E12.4 / 2E12.4, I4  )
C
      F        =      AO * SQRT ( ABS( 1.0 - AR**2) ) / 2.0
      XO1    =   X   - XI1
      XO2    =   X   - XI2
C
C                             CHECK ON INPUT FLAG
```

```
C                                  = 0   DZ ,  = 1   DY
C
      IF ( IDZDY .EQ. 1 )  GO TO 200
C                                                       * *         * *
C                                                       *    D Z      *
C                                                       * *         * *
      SGS    =   0.0
      CGS    =   1.0
      IF ( AR .LT. 1.0 )   GO TO 400
C
      Z01         =   Z  - ( ZETA + E )
      Z02         =   Z  - ( ZETA - E )
      Y01         =   Y  -    ETA
      Y02         =   Y01
      GO TO  300
C                                                       * *         * *
C                                                       *    D Y      *
C                                                       * *         * *
  200 SGS         =   -1.0
      IF  (LHS . NE . 0)   SGS = -SGS
      CGS         =    0.0
      IF ( AR .GT. 1.0 )   GO TO 400
C
      Z01         =   Z  -    ZETA
      Z02         =   Z01
      Y01         =   Y  - ( ETA + E)
      YC2         =   Y  - ( ETA - E)
C     GO TO 300                                   ****  DZ     AR    GE   1   ****
C                                                 ****  DY     AR    LE   1   ****
  300 CONTINUE
      L           =      0
      Z0          =      Z - ZETA
      Y0          =      Y -  ETA
C
      R1SQR       =      Y01**2   +   Z01**2
      R2SQR       =      Y02**2   +   Z02**2
      R1FOR       =      R1SQR**2
      R2FOR       =      F2SQR**2
C
      IF ( IPRNT .NE. 0 ) WRITE(NPOT,1000)X01,X02,Y01,YC2,Z01,Z02 ,
     *    SGR, CGR, SGS,
     *     CGS        , KR        , CBAR      , FMACH    , E         , L
C
      CALL      FLLD (X01     , X02     , Y01     , Z01     , SGR     , CGR     ,
     *                 SGS     , CGS     , KR      , CBAR    ,
     *                 FMACH,  E,   L,  KD1PR    , KD1PI    , KD2PR    , KD2PI      )
C
      IF ( AR .NE. 1.0 )   GO TO 320
C          IDENTICAL RESULTS FROM FLLD, THEREFORE SKIP SECOND CALL
      KD1MR       =   KD1PR
      KD1MI      =    KD1PI
      KD2MR      =    KD2PR
```

```
      KD2MI     =    KD2PI
      GO TO 360
  320 CONTINUE
      CALL FLLD     (X01    , X02    , Y02    , Z02    ,SGR    , CGR    ,
     *               SGS    , CGS    , KR     , CBAR   ,
     *               FMACH, E,  L, KD1MR   , KD1MI   , KD2MR   , KD2MI     )
      IF ( IPRNT .NE. 0)
     * WRITE ( NPOT, 1000)  KD1PR, KD1PI,  KD2PR,  KD2PI,
     *                 KD1MR, KD1MI, KD2MR, KD2MI
  360 CONTINUE
      DZDYR   = 0.0
      DZDYI   = 0.0
      IF  (R1SQR.LE.0.0001. OR .R2SQR.LE.0.0001)  GO TO  370
C             REAL
      DZDYR   = (   (KD1PR / R1SQR  + KD1MR / R2SQR  )
     *            + ( KD2PR / R1FOR  +  KD2MR / R2FOR  ) ) / PI16
     *     * ( -1.0 )
C             IMAGINARY
      DZDYI   = (   (KD1PI / R1SQR  + KD1MI / R2SQR  )
     *            + ( KD2PI / R1FOR  +  KD2MI / R2FOR  ) ) / PI16
     *     * ( -1.0 )
  370 CONTINUE
      IF ( IPRNT .NE. 0 )  WRITE ( NPOT,1000) DZDYR,DZDYI
C
C
      RETURN
C                            ****   DZ  -   AR  .LT.  1   ****
C                            ****   DY  -   AR  .GT.  1   ****
  400 SL1       =    0.0
      TL1       =    0.0
      SL2       =    0.0
      TL2       =    0.0
      CL1       =    1.0
      CL2       =    1.0
      E         =     1.732051 * E
      YO        =     Y - ETA
      ZO        =     Z - ZETA
C
      CALL  TVOR   ( SL1, CL1, TL1, SL2, CL2, TL2, SGS, CGS, SGR, CGR,
     *     X01    , X02   , YO    , ZO     , E      , BETA  , CBAR  ,
     *               FMACH , KR      ,
     *      DZDYR,    DZDYI,     IPRNT     )
      RETURN
 1000 FORMAT   ( 5F20.8/ 5E20.8 /       4E20.8, I10   )
      END
```

```
      SUBROUTINE  FLLD  ( X01, X02, Y0, Z0, SGR, CGR, SGS, CGS,  KR,
     *     CBAR    , FMACH , E      , L       , KD1R, KD1I  , KD2R   , KD2I   )
CFLLD          CALCULATION OF THE NUMERATOR OF A DOUBLET LINE OF FINITE
C     LENGTH.  LIKE KERN, THERE ARE TWO OUTPUT COMPLEX VALUES REPRESENT-
C              ED BY FOUR REAL NUMBERS AND AN INPUT OPTION.
C
C              WRITTEN BY D. H. LARSON, STRUCTURAL MECHANICS MDAC 11/70
C
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C *          X01   -      X - XI1                                        *
C *          X02   -      X - XI2                                        *
C *          Y0    -      Y - ETA                                        *
C *          Z0    -      Z - ZETA                                       *
C *          SGR   -      SIN ( GAMMA-R)                                 *
C *          CGR   -      COS ( GAMMA-R)                                 *
C *          SGS   -      SIN ( GAMMA-S)                                 *
C *          CGS   -      COS ( GAMMA-S)                                 *
C *          KR    -      REDUCED FREQUENCY                              *
C *          BR    -      REFERENCE LENGTH                               *
C *          FMACH-       MACH NUMBER                                    *
C *          E     -                                                     *
C *          L     -      OPTION FLAG USED IN TKER                       *
C *          KD1R  -      REAL PART OF  KD1                              *
C *          KD1I  -      IMAGINARY PART OF KD1                          *
C *          KD2R  -      REAL PART OF  KD2                              *
C *          KD2I  -      IMAGINARY PART OF KD2                          *
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      COMMON  /KDS /  IND, KK1R,    KK1I,  KK2R,  KK2I,  R1
      COMMON  /DLM/  K10, K20, K1RT1,  K1IT1, K2RT2P, K2IT2P,  K10T1,
     *                K20T2P,  E2
      REAL       KR,  KK1R,   KK1I,  KK2R,  KK2I
      REAL       KD1R,       KD1I ,     KD2R,        KD2I
      REAL   K10T1, K20T2P, K1RT1, K10, K2IT2P, K20, K2RT2P,K1IT1
      COMPLEX  KD1 , KD2, K1XI1, K1XI2, TEMP1, TEMP2, K2XI1, K2XI2
C
C             X01 =  X - XI1  AND  X02 = X - XI2, DELXI= XI2- XI1
      DELXI    =    X01 - X02
C
C                 FULL KERNEL FROM - TKER -
      IND      =    0
      KD1R     =    0.0
      KD2R     =    0.0
      E2       =    E **2
      T1        =      KR * DELXI / CBAR
      BF        =      CBAR / 2.0
      ST1      =    SIN ( T1 )
      CT1      =    COS ( T1 )
      I        =    1
      X0       =    X01
C
C
```

```
10 CALL  TKEP  ( XO    ,    YO    ,    ZO    ,    KR    ,    BR    ,      FLLD
  *     SGR  ,    CGR   ;    SGS   ,    CGS   ;    RT1   ;    RT2   ;      FLLD
  *     FMACH )                                                           FLLD
C                                                                         FLLD
      GO TO  (  30, 40), I                                                FLLD
30    K1XI1   =  CMPLX ( KK1R, KK1I )                                     FLLD
      K2XI1   =  CMPLX ( KK2R, KK2I )                                     FLLD
      IF ( L  .EQ.  0 )  GO TO 35                                         FLLD
      KD1R    =    KD1R  -  K1OT1                                         FLLD
      KD2R    =    KD2R  -  K2OT2P                                        FLLD
35    CONTINUE                                                            FLLD
C                                                                         FLLD
C          NOW GO CALCULATE FOR XI = XI2                                  FLLD
C                                                                         FLLD
      XO   =   XO2                                                        FLLD
      I    =   2                                                          FLLD
      GO TO  10                                                           FLLD
C                                                                         FLLD
40    K1XI2   =  CMPLX ( KK1R, KK1I )                                     FLLD
      K2XI2   =  CMPLX ( KK2R, KK2I )                                     FLLD
      IF ( L  .EQ. 0  )  GO TO 50                                         FLLD
      KD1R    =    KD1R  +  K1OT1                                         FLLD
      KD2R    =    KD2R  +  K2OT2P                                        FLLD
50    CONTINUE                                                            FLLD
C                                                                         FLLD
      TEMP1   =  CMPLX ( CT1,  ST1  )                                     FLLD
      TEMP2   =  CMPLX ( CT1, -ST1  )                                     FLLD
C                                                                         FLLD
C                DESIRED RESULTS  ( COMPLEX )                             FLLD
      KD1     =  K1XI1 * TEMP1  -  K1XI2 * TEMP2                          FLLD
      KD2     =     K2XI1 * TEMP1 - K2XI2 * TEMP2                         FLLD
C                                                                         FLLD
C                CONVERT TO REAL AND IMAGINARY PARTS                      FLLD
      KD1R    =  REAL ( KD1 )  +  KD1R                                    FLLD
      KD1I    =  AIMAG  ( KD1 )                                           FLLD
      KD2R    =   REAL ( KO2 )  +  KO2R                                   FLLD
      KD2I    =  AIMAG  ( KD2 )                                           FLLD
      RETURN                                                              FLLD
      END                                                                 FLLD
```

```
      SUBROUTINE  FWMW  (  NDX,   NE,      SGS,    CGS ,     IRB,     AO      ,
     *    ARB        , XBLE      , XBTE      , YB        , ZB        , XS        ,
     *    YS         , ZS        , NAS       , NASB      , KR        , BETA2     ,
     *    CBAR       , AVR       , FWZ,FWZA, FWY,FWYA, IF1,IPRNT, SCALER)
C   ***   CALCULATES THE EFFECT OF A DOUBLET PLUS ANY CONTRIBUTIONS
C         DUE TO IMAGES, SYMMETRY  AND GROUND EFFECT ON BODY
C  *                                                                 *
C  *      ********* *************************************************
C  *                                                                 *
C  *      NDX         NOT USED
C  *      NE          GROUND EFFECTS FLAG                            *
C  *      SGS         SINE OF SENDING POINT DIHEDRAL ANGLE           *
C  *      CGS         COSINE OF SENDING POINT DIHEDRAL ANGLE         *
C  *      IRB         NUMBER OF THE RECEIVING BODY                   *
C  *      AO          RADIUS OF THE BODY                             *
C  *      ARB         ARRAY OF RATIOS OF BODY AXIS                   *
C  *      XBLE        LEADING  EDGE COORDINATE OF SLENDER BODY ELEMENT
C  *      XBTE        TRAILING EDGE COORCINATE OF SLENDER BODY ELEMENT
C  *      YB          ARRAY CONTAINING THE Y-COORDINATES OF THE      *
C  *                  BODIES                                         *
C  *      ZB          ARRAY CCNTAINING THE Y-COORDINATES OF THE      *
C  *                  BODIES                                         *
C  *      XS          1/4-CHORD  X-COORDINATE  OF SLENDER BODY ELEMENT
C  *      YS          Y-COORDINATE OF SENDING POINT                  *
C  *      ZS          Z COORDINATE OF THE SENDING POINT              *
C  *      NAS         NUMBER OF ASSOCIATED BODIES                    *
C  *      NASB        ARRAY CCNTAINING THE ASSOCIATED BODY NOS.      *
C  *      KR          REDUCED FREQUENCY                              *
C  *      BETA2       = 1 - ( MACH ** 2 )
C  *      CBAR        REFERENCE CHARD LENGTH                         *
C  *      AVR         ARRAY CF BODY RADII                            *
C  *      FWZ, FWZA   OUTPUT Z-FORCES                                *
C  *      FWY, FWYA   OUTPUT Y FORCES                                *
C  *      IF1         FLAG INDICATING THE ORIENTATION OF THE         *
C  *                  RECEIVING BODY                                 *
C  *      IPRINT      PRINT FLAG
C  *                                                                 *
C  *      ********* *************************************************
C  *                                                                 *
      DIMENSION       YB      (  1  ), ZB     (  1  ), NASB  (  1  ),
     *    AVR     (  1   ),ARB    (  1  )
      COMPLEX    FWZ, FWY, FWZA, FWYA
      REAL          KR, MACH
      DATA    NPOT  /6/
      MACH      =   SQRT( 1.0 - BETA2 )
      FWZ       =   CMPLX(0.0, 0.0)
      FWZA      =   CMPLX(0.0, 0.0)
      FWY       =   CMPLX(0.0, 0.0)
      FWYA      =   CMPLX(0.0, 0.0)
C
      IF( IPRNT  .NE. 0 )  WRITE ( NPOT, 8000)
      DMMY      =   0.0
```

```
      INFL    =  1
C           ARG-R   ARGUMENTS
      DYB    =    YB ( IRB)
      DZB     =    ZB(IRB)
      DA    =    AO
      DAR         =      ARB ( IRB )
      DELEPS      =      1.0
      DELEP2      =      1.0
      C     =     CGS
      S    =    -SGS
      DY     =   YS
      DZ    =  ZS
      ITYPE   =  1
      IF2         =      0
      K           =      1
      ASSIGN   100   TO IRET1
      GO TO 2000
  100 SY    =  1.0
      SZ    =   1.0
      SG          =      SGS
      ASSIGN   200   TO IRET1
      GO TO 5000
  200 CONTINUE
      DELEPS      =      1.0
      DELEP2      =     -1.0
      C    =    CGS
      S    =    SGS
      DY    =    -YS
      DZ     =     ZS
      ITYPE    =  1
      IF2         =      0
      K           =      2
C      IF SENDING ELEMENT LIES ON THE Z  AXIS.. DON'T USE SYMMETRY
      IF   (  DY  .EQ.  0.0 )  GO TO 700
      ASSIGN  300   TO  IRET1
      GO  TO  2000
  300 CONTINUE
      SY    =  -1.0
      SZ    =  1.0
      SG          =      -SGS
      ASSIGN  400   TO  IRET1
      GO TO 5000
  400 CONTINUE
C                CHECK GROUND EFFECTS FLAG--SKIP IF ZERO
      IF   (  NE  .EQ.  0 )  GO TO 7000
C                PORTION FOR COMBINATION OF SYMMETRY AND GROUND EFFECTS
      ITYPE   =  1
      IF2         =      0
      K           =      3
      DELEPS      =      NE
      DELEP2      =     -NE
      C           =      CGS
```

```
      S          =     -SGS
      DY     =   -YS
      DZ    =    -ZS
C     IF THE SENDING ELEMENT LIES ON THE Z OR Y AXIS DON'T USE GRND.
C                                 OR SYMMETRY
      IF  (  DZ  .EQ.  0.0) GO TO 7000
      ASSIGN  500   TO IRET1
      GO  TO  2000
  500 CONTINUE
      SY     =   -1.0
      SG         =     SGS
      SZ   =   -1.0
      ASSIGN  600   TO IRET1
      GO  TO  5000
  600 CONTINUE
      GO TO 800
C                SKIP GROUND EFFECTS CALCULATIONS IF FLAG IS ZERO
  700 IF   (  NE  .EQ.  0 )  GO TO 7000
C                PORTION FOR GROUND EFFECTS ONLY
  800 CONTINUE
      DELEPS     =     NE
      DELEP2     =     NE
      DY   =   YS
      DZ    =     -ZS
C      IF SENDING ELEMENT LIES ON THE Y AXIS DON'T USE GROUND EFFECTS
      IF  (  DZ  .EQ.  0.0) GO TO 7000
      C          =     CGS
      S          =     SGS
      ITYPE    =  1
      IF2        =     0
      K          =     4
      ASSIGN  900   TO IRET1
      GO TO 2000
  900 CONTINUE
      SY     =   1.0
      SZ      =   -1.0
      SG         =     -SGS
      ASSIGN  1000 TO IRET1
      GO TO 5000
 1000 CONTINUE
 1100 RETURN
C
C               CALCULATION OF EFFECTIVE FORCES
C
 2000 CONTINUE
C
      CALL FZY2    ( XS       , XBLE      , XBTE     , DY       , DZ      ,
     *    DYB      , DZB      , DA        , BETA2    , CBAR     , KR      ,
     *    DFZZR    , DFZZI    , DFZYR     , DFZYI    , DFYZR    , DFYZI   ,
     *    DFYYR    , DFYYI    , IPRNT                                     )
      IF ( IPRNT .NE. 0 )  WRITE ( NPOT , 8010 )
     *   XS,  XBLE,  XBTE,  DY,  DZ,  DYB,  DZB,  DA,
```

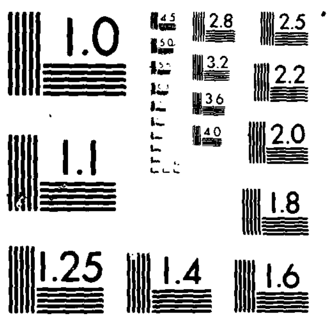
```
     *   BETA2, DFZZR, DFZZI, DFZYR, DFZYI, DFYYR, DFYYI
C
      FWZR      =     C * DFZZR    +  S * DFZYR
      FWZI      =     C * DFZZI    +  S * DFZYI
      FWZ       =     FWZ  +  DELEPS * CMPLX ( FWZR,  FWZI )
      FWZA      =     FWZA +  DELEP2 * CMPLX ( FWZR,  FWZI )
      FWYR      =     C * DFYZR    +  S * DFYYR
      FWYI      =     C * DFYZI    +  S * DFYYI
      FWY       =     FWY  +  DELEPS * CMPLX ( FWYR,  FWYI )
      FWYA      =     FWYA +  DELEP2 * CMPLX ( FWYR,  FWYI )
      GO TO ( 3000, 6000) , ITYPE
 3000 GO TO IRETL, ( 100,  200,  300,  400,  500,  600,  800,  900,
     *               1000 )
C
C               CALCULATION LOOP FOR ASSOCIATED BODIES
C
 5000 IF (  NAS  .LE. 0 ) GO TO 3000
      I  =  1
      ITYPE    =  2
 5100 IB  =  NASB(I)
C               CHECK TO SEE IF THE ASSOCIATED BODY IS THE RECEIVING
C               BODY.
      IF2        =     0
      IF (  IB  .NE.  IRB )  GO TO 5800
C               IF IT IS DETERMINE IF THE SENDING POINT IS OUTSIDE
C               OR INSIDE THE BODY.
      GO  TO  (  5600,  5500,  5400,  5300 ) , K
 5300 IF (  DYB  .NE. 0.0 )  GO TO 5800
      IF2        =     1
      GO TO 5800
 5400 IF ( DYB .NE. 0.0 )  GO TO 5800
 5500 IF (  DZB  .NE.  0.0  )  GO TO 5800
 5600 IF2        =     1
 5800 CONTINUE
      ETA       =     SY * YS
      ZETA      =     SZ * ZS
      ZBI       =     SZ  *  ZB(IB)
      YBI       =     SY  *  YB(IB)
      DARIB     =     ARB ( IB )
      DAIB      =     AVR ( IB )
      TEST      =    SQRT((ETA-YBI)**2 + (ZETA-ZBI)**2)
      IF  (TEST .GT. SCALER  *  DAIB)  GO TO  6000
      CALL  SUBI  ( DAIB    , ZBI     , YBI     , DARIB    ,
     *                ETA    ,  ZETA   ,  CGS    ,  SG     ,
     *                DMMY   ,  DMMY   ,  DMMY   ,  DY     ,  DZ     ,
     *                DMMY   ,  DMMY   ,  DMMY   ,  DMMY   ,  S      ,
     *                C      ,  INFL   ,  IOUTFL   )
      IF ( IPRNT  .NE.  0 )      WRITE ( NPOT , 8020 )
     *   DAIB,  ZBI,  YBI,  DARIB,  ETA,  ZETA,  CGS,  SG,  DY,
     *          DZ,   S,    C,     INFL,   IOUTFL
      IF ( IOUTFL ) 2000, 3000, 2000
 6000 I = I + 1
```

```
      IF ( I - NAS ) 5100,  5100,  3000                                FWMW 210
 7000 RETURN                                                           FWMW 211
 8000 FORMAT ( 14HOENTERING FWMW )                                     FWMW 212
 8010 FORMAT (  5(  5E20.8/) / 2I10  )                                 FWMW 213
 8020 FORMAT (  7H SUBI../  5E20.8/  5E20.8/  2E20.8,2I20      )       FWMW 214
      END                                                              FWMW 215
```

AD-A106 520 DOUGLAS AIRCRAFT CO LONG BEACH CA F/G 18/3
NUCLEAR BLAST RESPONSE COMPUTER PROGRAM, VOLUME III. PROGRAM LI--ETC(U)
AUG 81 J A MCGREW, H H CROXEN, T P KALMAN DNA001-75-C-0216

UNCLASSIFIED AFWL-TR-81-32-VOL-3 NL

2 of 5
AD
A106520



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

```
      SUBROUTINE FZY2 (XIJ, X1, X2,ETA,ZETA, YB, ZB, A, BETA2,CBAR, K,
     1              FZZR, FZZI, FZYR, FZYI, FYZR, FYZI, FYYR, FYYI, MFLG )
C    ***     THIS SUBROUTINE IS AN ALTERNATIVE TO SUBROUTINE  FMZY   ---
C            IT IS USED WHENEVER THE OPTION FLAG   IBFS  =  1
C    ***
      REAL  M,K,KBAR,KBAR2,I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,KBAR3
      DATA      M,      KBAR, KBAR2, LASTBR / 3*0.0, 0 /
      DATA      TEST1,TEST2,CTH,STH,RAIJ,RAIJ2 /0.142857, 0.5, 1.0,3*0.0/
      DATA      CAPDR,CAPDI,I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11/ 13*0.0 /
   10 FORMAT   (1H0, 6E20.8/)
   20 FORMAT   (1H0,10X, 12HCAPDR, CAPDI/)
   30 FORMAT   (1H0//10X, 4HXI =,F12.6, 6H  X1 =,F12.6,6H  X2 =,F12.6/
     1           10X, 5HETA =,F11.6, 8H  ZETA =,F10.6,6H  YB =,F12.6,
     2           6H  ZB =, F12.6/ 10X,5HA    =,F11.6, 8H  BETA2=, F10.6,
     3           8H  CBAR =, F12.6//)
   40 FORMAT   (1H0,10X, 35HM ,KBAR, XA, DX, A2, RAIJ, CTH, STH/)
   50 FORMAT   (1H0,10X, 6HFZZR =, E20.8, 8H  FZZI =, E20.8/  9X,
     1                    8H  FZYR =, E20.8, 8H  FZYI =, E20.8/  9X,
     2                    8H  FYYR =, E20.8, 8H  FYYI =, E20.8/)
   60 FORMAT   (1H0,10X, 36HRWIG, RAA, CAPA, EARG, QR, QI, DELTA/)
   70 FORMAT   (1H0,10X, 43HI1, I2, I3, I4, I5, I6, I7, I8, I9, I10,I11/)
   80 FORMAT   (1H0, 10X,  3HTAU/)
   90 FORMAT   (1H0, 10X,  8HRA1, RA2/)
  100 FORMAT   (1H0,10X, 20HFTHR, FTHI, FRR, FRI/)
      IF  (MFLG.NE.0) WRITE (6,30) XIJ,X1,X2,ETA,ZETA,YB,ZB,A,BETA2,CBAR
      M       = SQRT(1.0 - BETA2)
      IF   (K. LE .0.0001 . AND . M. LE .0.0001)  GO TO  110
      KBAR    = 2.0 *K *M *A / CBAR
      KBAR2   = KBAR*KBAR
      GO TO  120
  110 CONTINUE
      KBAR    = 0.0
      KBAR2   = 0.0
  120 XA      = 0.5 * (X1 + X2)
      DX      = X2 - X1
      A2      = A * A
      EPS     = 0.001 * A2
      IF   (ETA. EQ .YB. AND . ZETA. EQ .ZB)  GO TO  130
      RAIJ2   = (ETA-YB)**2 + (ZETA-ZB)**2
      RAIJ    = SQRT(RAIJ2)
      CTH     = (ETA- YB) / RAIJ
      STH     = (ZETA-ZB) / RAIJ
      IF   (RAIJ2. GT .A2)  GO TO  150
      GO TO  140
  130 CONTINUE
      RAIJ    = 0.0
      RAIJ2   = 0.0
      CTH     = 1.0
      STH     = 0.0
  140 RWIG2   = A2
      GO TO 160
  150 RWIG2   = RAIJ2
```

```
160 RAA     = SQRT((XA -XIJ)**2 + BETA2*RWIG2)
    CT2     = CTH*CTH
    ST2     = 0.0
    IF   (ABS(STH) . GT . 0.0001)  ST2=STH*STH
    RWIG    = SQRT(RWIG2)
    IF   (MFLG.EQ.0)  GO TO  170
    WRITE   (6,40)
    WRITE   (6,10)   M, KBAR, XA, DX, A2, RAIJ, CTH, STH
170 CONTINUE
    RAA2    = RAA * RAA
    RAA3    = RAA * RAA2
    RAA4    = RAA * RAA3
    CAPA    = M - (XA- XIJ) / RAA
    DELTA   = DX / RAA
    DELTA2  = DELTA * DELTA
    EARG    = 0.0
    IF   (KBAR . LE . 0.0001)  GO TO  180
    EARG    = KBAR * (M * (XA-XIJ) - RAA) / (BETA2 * A)
    QR      = COS(EARG) / (4.0 * DX)
    QI      = SIN(EARG) / (4.0 * DX)
    GO TO  190
180 QR      = 1.0 / (4.0 * DX)
    QI      = 0.0
190 CONTINUE
    IF   (MFLG.EQ.0)  GO TO  200
    WRITE   (6,60)
    WRITE   (6,10)   RWIG, RAA, CAPA, EARG, QR, QI, DELTA
200 CONTINUE
    IF   (DELTA. GT . TEST1)  GO TO  240
    I1      = DELTA / RAA2
    TRM1    = BETA2 * A * I1
    FTHR    = A * QR * TRM1
    FTHI    = A * QI * TRM1
    IF   (KBAR. LE .0.0001)  GO TO  210
    I4      = DELTA / RAA
    TRM2    = KBAR * I4
    FTHR    = FTHR - A * QI * TRM2
    FTHI    = FTHI + A * QR * TRM2
210 CONTINUE
    IF   (RAIJ2. GT . (A2+EPS))  GO TO  220
    FRR     = FTHR
    FRI     = FTHI
    GO TO  370
220 I6      = DELTA / RAA4
    TRM1    = -3.0 * A2 * BETA2*BETA2 * I6
    CAPDR   = RAIJ2 * QR * TRM1
    CAPDI   = RAIJ2 * QI * TRM1
    IF   (KBAR . LE . 0.0001)  GO TO  230
    I9      = DELTA / RAA3
    TRM1    = TRM1 + KBAR2 * I1
    TRM2    = -3.0 * A * BETA2 * KBAR * I9
    CAPDR   = RAIJ2 * (QR * TRM1 - QI * TRM2)
```

```
      CAPDI   = RAIJ2 * (QR * TRM2 + QI * TRM1)                         FZY2 106
  230 FRR     = FTHR + CAPDR                                            FZY2 107
      FRI     = FTHI + CAPDI                                            FZY2 108
      GO TO  370                                                        FZY2 109
  240 CONTINUE                                                          FZY2 110
      IF   (DELTA . GT . TEST2)  GO TO  320                             FZY2 111
      LASTBR = 0                                                        FZY2 112
      TAU     = (XA - XIJ) / RAA                                        FZY2 113
      TAU2    = TAU * TAU                                               FZY2 114
      I1      = DELTA * (1.0 - (-1.0+5.0*TAU2)*DELTA2/8.0) / RAA2       FZY2 115
  250 TRM1    = A * BETA2 * I1                                          FZY2 116
      FTHR    = A * QR * TRM1                                           FZY2 117
      FTHI    = A * QI * TRM1                                           FZY2 118
      IF   (KBAR . LE . 0.0001)  GO TO  270                             FZY2 119
      IF   (LASTBR . NE . 0)  GO TO  350                                FZY2 120
      DELTA3 = DELTA * DELTA2                                           FZY2 121
      I2      = -(TAU * DELTA3) / (4.0 * RAA)                           FZY2 122
      I3      = DELTA3 / 12.0                                           FZY2 123
      I4      = DELTA * (1.0 + (-1.0+3.0*TAU2)*DELTA2/12.0) / RAA       FZY2 124
      I5      = -(TAU * DELTA3) / 6.0                                   FZY2 125
  260 TRM1    = TRM1 - (KBAR2 * CAPA * I5) / (A * BETA2)                FZY2 126
      TRM2    = KBAR * (CAPA * I2 + I4 - I3*BETA2*RWIG2/(2.0*RAA3) )    FZY2 127
      FTHR    = A * (QR * TRM1 - QI * TRM2)                             FZY2 128
      FTHI    = A * (QR * TRM2 + QI * TRM1)                             FZY2 129
  270 IF   (RAIJ2. GT . (A2+EPS))  GO TO  280                           FZY2 130
      FRR     = FTHR                                                    FZY2 131
      FRI     = FTHI                                                    FZY2 132
      GO TO  370                                                        FZY2 133
  280 CONTINUE                                                          FZY2 134
      KBAR3   = KBAR*KBAR2                                              FZY2 135
      IF   (LASTBR . NE . 0)  GO TO  340                                FZY2 136
      I6      = DELTA * (1.0 + 5.0*(-1.0+7.0*TAU2)*DELTA2/24.0) / RAA4  FZY2 137
  290 TRM1    = -3.0 * A2 * BETA2*BETA2 * I6                            FZY2 138
      CAPDR   = RAIJ2 * QR * TRM1                                       FZY2 139
      CAPDI   = RAIJ2 * QI * TRM1                                       FZY2 140
      IF   (KBAR . LE . 0.0001)  GO TO  310                             FZY2 141
      IF   (LASTBR . NE . 0)  GO TO  360                                FZY2 142
      I7      = -5.0 * TAU * DELTA3 / (12.0 * RAA3)                     FZY2 143
      I8      = DELTA3 / (12.0 * RAA2)                                  FZY2 144
      I9      = DELTA * (1.0 + (-1.0+6.0*TAU2)*DELTA2/6.0) / RAA3       FZY2 145
      I10     = -DELTA3 * TAU / (3.0 * RAA2)                            FZY2 146
  300 TRM1    = TRM1 + KBAR2 * (I1 + 3.0 * CAPA * I10)                  FZY2 147
      TRM2    = 3.0*A*BETA2 * KBAR * (-CAPA*I7 +I8*BETA2*RWIG2/(2.0*RAA3)FZY2 148
     1        9) + KBAR3 * CAPA * I2 / (A * BETA2)                      FZY2 149
      CAPDR   = RAIJ2 * (QR * TRM1 - QI * TRM2)                         FZY2 150
      CAPDI   = RAIJ2 * (QR * TRM2 + QI * TRM1)                         FZY2 151
  310 FRR     = FTHR + CAPDR                                            FZY2 152
      FRI     = FTHI + CAPDI                                            FZY2 153
      GO TO  370                                                        FZY2 154
  320 CONTINUE                                                          FZY2 155
      LASTBR = 1                                                        FZY2 156
      RWIG    = SQRT(RWIG2)                                             FZY2 157
```

```
      RA12   = (X1 - XIJ)**2 + BETA2 * RWIG2                         FZY2 158
      RA22   = (X2 - XIJ)**2 + BETA2 * RWIG2                         FZY2 159
      RA1    = SQRT(RA12)                                            FZY2 160
      RA2    = SQRT(RA22)                                            FZY2 161
      IF  (MFLG.EQ.0)  GO TO  330                                    FZY2 162
      WRITE  (6,90)                                                  FZY2 163
      WRITE  (6,10)  RA1, RA2                                        FZY2 164
  330 CONTINUE                                                       FZY2 165
      I1     = ((X2-XIJ)/RA2 - (X1-XIJ)/RA1) / (BETA2*RWIG2)         FZY2 166
      GO TO  250                                                     FZY2 167
  340 CONTINUE                                                       FZY2 168
      RA13   = RA1 * RA12                                            FZY2 169
      RA23   = RA2 * RA22                                            FZY2 170
      I6     = ((X2-XIJ)/RA23-(X1-XIJ)/RA13 + 2.0*I1)/(3.0*BETA2*RWIG2) FZY2 171
      GO TO  290                                                     FZY2 172
  350 PART1  = 0.5 * DX * (XA - XIJ)                                 FZY2 173
      I2     = -((PART1+RAA2)/RA2 + (PART1-RAA2)/RA1)/ (BETA2*RWIG2) FZY2 174
      DENOM  = X1 - XIJ + RA1                                        FZY2 175
      IF  (ABS(DENOM) . LE . 0.0001)  STOP                           FZY2 176
      I11    = ALOG(ABS((X2 - XIJ + RA2) / DENOM))                   FZY2 177
      I3     = I11 - 2.0*(XA - XIJ)*I2 - RAA2 * I1                   FZY2 178
      DENO4  = SQRT(BETA2) * RWIG                                    FZY2 179
      ARG1   = (X2 - XIJ) / DENO4                                    FZY2 180
      APG2   = (X1 - XIJ) / DENO4                                    FZY2 181
      I4     = (ATAN(ARG1) - ATAN(ARG2)) / DENO4                     FZY2 182
      I5     = 0.5 * ALCG(RA22 / RA12) - (XA - XIJ) * I4             FZY2 183
      GO TO  260                                                     FZY2 184
  360 CONTINUE                                                       FZY2 185
      I7     = -(1.0/RA23 - 1.0/RA13) / 3.0 - (XA - XIJ) * I6        FZY2 186
      I8     = I1 - 2.0 * (XA - XIJ) * I7 - RAA2 * I6                FZY2 187
      I9     = ((X2-XIJ)/RA22-(X1-XIJ)/RA12 + I4) / (2.0*BETA2*RWIG2) FZY2 188
      I10    = -((PART1 + RAA2)/RA22 + (PART1 - RAA2)/RA12 +         FZY2 189
     1          (XA - XIJ) * I4) / (2.0 * BETA2 * RWIG2)             FZY2 190
      GO TO 300                                                      FZY2 191
  370 CONTINUE                                                       FZY2 192
      IF  (RAIJ2. LF .(A2-EPS) . OR . MFLG.EQ .0)  GO TO 380         FZY2 193
      WRITE  (6,20)                                                  FZY2 194
      WRITE  (6,10)  CAPDR, CAPDI                                    FZY2 195
  380 CONTINUE                                                       FZY2 196
      IF  (MFLG.EQ.0)  GO TO  390                                    FZY2 197
      WRITE  (6,70)                                                  FZY2 198
      WRITE  (6,10)  I1, I2, I3, I4, I5, I6, I7, I8, I9, I10, I11    FZY2 199
      WRITE  (6,100)                                                 FZY2 200
      WRITE  (6,10)  FTHR, FTHI, FRR, FRI                            FZY2 201
  390 CONTINUE                                                       FZY2 202
      FZZR   = CT2 * FTHR + ST2 * FRR                                FZY2 203
      FZZI   = CT2 * FTHI + ST2 * FRI                                FZY2 204
      FYYR   = ST2 * FTHP + CT2 * FRR                                FZY2 205
      FYYI   = ST2 * FTHI + CT2 * FRI                                FZY2 206
      IF  (CTH. EQ .0.0 . CR . STH. EQ . 0.0)  GO TO  400            FZY2 207
      FZYR   = CTH * STH * (FRR - FTHR)                              FZY2 208
      FZYI   = CTH * STH * (FRI - FTHI)                              FZY2 209
```

```
      GO TO  410                                                        FZY2 210
  400 FZYR    = 0.0                                                     FZY2 211
      FZYI    = 0.0                                                     FZY2 212
  410 CONTINUE                                                          FZY2 213
      IF   (MFLG.NE.0)  WRITE (6,50)  FZZR,FZZI,FZYR,FZYI,FYYR,FYYI     FZY2 214
      FYZR = FZYR                                                       FZY2 215
      FYZI = FZYI                                                       FZY2 216
      RETURN                                                            FZY2 217
      END                                                               FZY2 218
```

```
      SUBROUTINE GEND(NPRINT, NTAPE, NTOT, IOPT, MASTDT, DT, DPZ, DPY,  GEND   2
     1   DTA, DPZA, DPYA                                                GEND   3
     2 , IFLA, NBEA, NT12, NBARAY, NCARAY, YB, ZB, ARB, AVR, RIA, XLE   GEND   4
     3 , XTE, TH1A, TH2A, X, CG, EE, SG, YS, ZS, XIC, YIN, ZIN, DELX    GEND   5
     4 , XLAM, SCALER )                                                 GEND   6
      DIMENSION IFLA(2,1), NBEA(2,1), NT12(2,1)                         GEND   7
      DIMENSION NBARAY(1), NCARAY(1), YB(1), ZB(1), ARB(1),AVR(1),RIA(1)GEND   8
      DIMENSION XLE(1), XTE(1), TH1A(1), TH2A(1), X(1), CG(1), EE(1)    GEND   9
      DIMENSION SG(1), YS(1), ZS(1), XIC(1), YIN(1), ZIN(1), DELX(1)    GEND  10
      DIMENSION XLAM(1)                                                 GEND  11
C  ***   GENERATES THE INFLUENCE COEFFICIENT MATRIX  DT   USING THE     GEND  12
C        FOLLOWING FOUR SUBROUTINES  --  DPPS, DPZY, DZPY,  AND  DYPZ   GEND  13
      COMMON /AROCOM/ NTI, MODES                                        AROCOM 2
     X , NP, MSTRIP, NSMAX, NCMAX, NTOTAL, NB, MSBE, MBE                AROCOM 3
     Y , ND, NE, NBY, NBZ, NTO, NTP, NTY, NTZ                           AROCOM 4
     1 , NTYS, NTZS, MAXGR, MAXSTR, NSBETO, NSTRIP, KR, XM, REFA, REFC  AROCOM 5
     2 , REFS, FMACH, LINES                                             AROCOM 6
      DIMENSION  NTAPE(20)                                              GEND  15
      COMPLEX    DT( 1), DPZ( 1), DPY( 1), DTA( 1), DPZA( 1), DPYA( 1)  GEND  16
      REAL       KR                                                     GEND  17
   10 FORMAT  ( 1H1 /// )                                               GEND  18
   20 FORMAT  ( /// 10X, 42H     SYMMETRIC  -DT- MATRIX   **   ROW NO. , GEND  19
     1    I4,    6H  WITH , I4, 10H  ELEMENTS  / )                      GEND  20
   25 FORMAT  ( /// 10X, 42HANTISYMMETRIC  -DT- MATRIX   **   ROW NO. , GEND  21
     1    I4,    6H  WITH , I4, 10H  ELEMENTS  / )                      GEND  22
   30 FORMAT  (   6E16.6 )                                              GEND  23
      ITP1  = NTAPE(1)                                                  GEND  24
      ITP2  = NTAPE(2)                                                  GEND  25
      ITP3  = NTAPE(3)                                                  GEND  26
      ITP4  = NTAPE(4)                                                  GEND  27
      ITP8  = NTAPE(8)                                                  GEND  28
      REWIND  ITP1                                                      GEND  29
      REWIND  ITP2                                                      GEND  30
      REWIND  ITP3                                                      GEND  31
      REWIND  ITP4                                                      GEND  32
      REWIND  ITP8                                                      GEND  33
      EPS   = 0.00001                                                   GEND  34
      PI    = 3.1415926                                                 GEND  35
      FLND  = FLOAT(ND)                                                 GEND  36
      FLNE  = FLOAT(NE)                                                 GEND  37
      BETA  = SQRT(1.0-FMACH**2)                                        GEND  38
      FL    = REFC                                                      GEND  39
      NPV   = 0                                                         GEND  40
      NBOX  = NTP                                                       GEND  41
      LBO   = 1                                                         GEND  42
      LSO   = 1                                                         GEND  43
      JBO   = 1                                                         GEND  44
      KB    = 0                                                         GEND  45
      KT    = 0                                                         GEND  46
      DP    = (0.0,0.0)                                                 GEND  47
      DO  40  I=1,500                                                   GEND  48
      DT(I) = (0.0,0.0)                                                 GEND  49
```

```
   40 CONTINUE                                                          GEND
      I1     = 1                                                        GEND
      I2     = NTP                                                      GEND
      J1     = 1                                                        GEND
      J2     = NTP                                                      GEND
                                                                        GEND
C     DPP-LOOP                                                          GEND
      K      = 1                                                        GEND
C     K IS THE PANEL NUMBER ASSOCIATED WITH RECEIVING POINT  I          GEND
      KS     = 1                                                        GEND
C     KS IS THE STRIP NUMBER ASSOCIATED WITH RECEIVING POINT  I         GEND
      NBXR = NCARAY(K)                                                  GEND
      DO  60  I=I1,I2                                                   GEND
      SGR    = SG(KS)                                                   GEND
      CGR    = CG(KS)                                                   GEND
      CALL        DPPS(K,KS,I,J1,J2,SGR,CGR,REFC,FMACH,YS,ZS,NBARAY,    GEND
     1   NCARAY,DT,DTA                                                  GEND
     2 , YB,ZB,ARB,AVR,XLE,XTE,X,CG,FE,SG,XIC,DELX,XLAM,  SCALER )      GEND
      WRITE  (ITP1)  (DT(J), J=J1,J2), (DTA(J), J=J1,J2)                GEND
      IF  (I.EQ.I2)   GO TO  60                                         GEND
      IF  (I.EQ.NBARAY(K))   K=K+1                                      GEND
      IF  (I.EQ.NBXR)   GO TO    50                                     GEND
      GO TO  60                                                         GEND
   50 CONTINUE                                                          GEND
      KS     = KS+1                                                     GEND
      NBXR = NBXR+NCARAY(K)                                             GEND
   60 CONTINUE                                                          GEND
      NSTRIP = KS                                                       GEND
      NZYSV= 0                                                          GEND
      DO  70   J=J1,J2                                                  GEND
      DT(J)= (0.0,0.0)                                                  GEND
   70 CONTINUE                                                          GEND
      IF  (NB.EQ.0)   GO TO  320                                        GEND
      IF  (NTZ.EQ.0)   GO TO  180                                       GEND
      I1     = I2+1                                                     GEND
      I2     = I2+NTZ                                                   GEND
                                                                        GEND
C     DPZ-LOOP      **     ALSO USED FOR GENERATING THE DPY-MATRIX  --  SEE GEND
C     COMMENT IN  DPY-LOOP  BELOW                                       GEND
   80 CONTINUE                                                          GEND
      KB     = KB+1                                                     GEND
C     KB  IS THE BODY NUMBER ASSOCIATED WITH RECEIVING POINT  I         GEND
      IZ     = 0                                                        GEND
      KT     = KT+1                                                     GEND
C     KT  IS THE INDEX OF THE ARRAY OF FIRST-AND-LAST-ELEMENTS FOR THETA-1 GEND
C     FOR ALL BODIES  (ARRAY NAME IS  IFLA(100,2))                      GEND
      ICOUNT = 1                                                        GEND
      IFL    = 1                                                        GEND
      NZYKB = NBEA(2,KB)                                                GEND
      IFIRST = IFLA(1,KT)                                               GEND
      ILAST  = IFLA(2,KT)                                               GEND
      DO  170  I=I1,I2                                                  GEND
```

```
      DO  90  J=J1,J2
      DPZ(J) =  (0.0,0.0)
      DPY(J) =  (0.0,0.0)
      DPZA(J)=  (0.0,0.0)
      DPYA(J)=  (0.0,0.0)
   90 CONTINUE
      CALL         DPZY(   KB,IZ,I,J1,J2,IFIRST,ILAST,REFC,FMACH,YB,ZB,
     1             AVR,ARB,THIA,TH2A,NT12,NBARAY,NCARAY,NZYKB,DPZ,DPY,
     2   DPZA,DPYA
     3 , XLE  , XTE  , X, CG, EE, SG, YS, ZS, XIC, DELX, XLAM, SCALER )
      GO TO  (100,100,110),   NZYKB
  100 CONTINUE
      WRITE   (ITP1)   (DPZ(J), J=J1,J2), (DPZA(J), J=J1,J2)
      IF  (NZYKB.EQ.1)   GO TO 120
  110 CONTINUE
      WRITE   (ITP4)   (DPY(J), J=J1,J2), (DPYA(J), J=J1,J2)
  120 CONTINUE
      IF  (IZ.EQ.NBEA(1,KB)) GO TO 130
      IF  (IZ.EQ.ILAST.AND.ICOUNT.LT.IFL)  GO TO 160
      GO TO  170
  130 CONTINUE
      IZ      = 0
      IF  (NZYSV.LE.1.AND.NZYKB.GE.2)  GO TO  140
      GO TO  150
  140 CONTINUE
      LBO   = KB
      LSO   = NSTRIP+LBO
      JBO = I-NBEA(1,KB)-NBOX+1
  150 CONTINUE
      NZYSV = NZYKB
      IF   (I . EQ . I2)  GO TO  180
      KB      = KB+1
      ICOUNT = 0
      IFL     = 1
      NZYKB = NBEA(2,KB)
  160 CONTINUE
      KT      = KT+1
      ICOUNT = ICOUNT+1
      IFIRST = IFLA(1,KT)
      ILAST = IFLA(2,KT)
  170 CONTINUE
  180 CONTINUE
      IF  (I2.EQ.NTO)  GO TO  190

C   DPY-LOOP     **     THIS LOOP IS REDUCED TO SETTING THE CORRECT INDICES
C   AND USING THE   DPZ-LOOP  ABOVE
      I1    = I2+1
      I2   = NTO
      GO TO 80
  190 CONTINUE
      I1     = 1
      I2     = NTP
```

```
      IF (NTZ.EQ.0)  GO TO  250

C   DZP-LOOP
      K     = 1
C   K  IS THE PANEL NUMBER ASSOCIATED WITH RECEIVING POINT  I
      KS    = 1
C   KS  IS THE STRIP NUMBER ASSOCIATED WITH RECEIVING POINT  I
      NBXR  = NCARAY(K)
      KB    = 0
C   HERE  KB=0  SERVES AS A FLAG INDICATING THAT THE RECEIVING POINT   I
C   IS ON A PANEL AND NOT   ON A BODY
      J1    = J2+1
      J2    = J2+NTZ
      DO  210  I=I1,I2
      LS    = NSTRIP+1
      SGR   = SG(KS)
      CGR   = CG(KS)
      CALL        DZPY(KB,KS,LS,IZ,I,J1,J2,NYFLAG,FLND,FLNE,SGR,CGR,REFC,
     1    FMACH,KR,ARB,NBEA,DT,DTA
     2  , YB, ZB, RIA, X, YS, ZS, DELX )
      WRITE  (ITP2)   (DT(J), J=J1,J2), (DTA(J), J=J1,J2)
      IF (I.EQ.I2)  GO TO  210
      IF (I.EQ.NBARAY(K))  K =K +1
      IF (I.EQ.NBXR)  GO TO  200
      GO TO  210
  200 CONTINUE
      KS    = KS+1
      NBXR  = NBXR+NCARAY(K)
  210 CONTINUE

      NYFLAG = 0
C   DZZ-LOOP      **     ALSO USED FOR GENERATING THE  DZY  MATRIX  --  SEE
C   COMMENT IN  DZY-LOOP  BELOW
      KB    = 1
C   KB  IS THE BODY NUMBER ASSOCIATED WITH RECEIVING POINT  I
      KS    = NSTRIP+1
      IZ    = 0
      I1    = I2+1
      I2    = I2+NTZ
      SGR   = 0.0
      CGR   = 1.0
  220 CONTINUE
      LS    = NSTRIP+1
      LSX   = LS
      DO  240  I=I1,I2
      LS    = LSX
      IZ    = IZ+1
C   KS IS THE INDEX OF THE Y  AND  Z  COORDINATES OF RECEIVING POINT I
C   IN THE  DZZ-LOOP  KS RUNS FROM  (NSTRIP+1)  THROUGH  (NSTRIP+NBZ)
C   IN THE  DZY-LOOP  KS  RUNS FROM  (NSTRIP+NB-NBY+1) THROUGH  NSTRIP+NB
      CALL        DZPY(KB,KS,LS,IZ,I,J1,J2,NYFLAG,FLND,FLNE,SGR,CGR,REFC,
     1    FMACH,KR,ARB,NBEA,DT,DTA
```

```
     2 , YB, ZB, RIA, X, YS, ZS, DELX )                                 GEND
       WRITE  (ITP2)  (DT(J), J=J1,J2), (DTA(J), J=J1,J2)               GEND
       IF  (IZ.EQ.NBEA(1,KB)) GO TO 230                                  GEND
       GO TO  240                                                        GEND
   230 CONTINUE                                                          GEND
       IZ     = 0                                                        GEND
       KB     = KB+1                                                     GEND
       KS     = KS+1                                                     GEND
   240 CONTINUE                                                          GEND
       IF  (NTY.EQ.0)  GO TO 320                                         GEND
       IF  (NYFLAG.NE.0)  GO TO  250                                     GEND
                                                                         GEND
C   DZY-LOOP     **    THIS LOOP IS  REDUCED TO SETTING THE CORRECT INDICESGEND
C   AND USING THE  DZZ-LOOP  ABOVE                                       GEND
       I1     = NTO-NTY+1                                                GEND
       I2     = NTO                                                      GEND
       NYFLAG = 1                                                        GEND
       KB     = LBO                                                      GEND
       KS     = LSO                                                      GEND
       SGR    =-1.0                                                      GEND
       CGR    = 0.0                                                      GEND
       GO TO  220                                                        GEND
   250 CONTINUE                                                          GEND
       IF  (NTY.EQ.0)  GO TO 320                                         GEND
       I1     = 1                                                        GEND
       I2     = NTP                                                      GEND
                                                                         GEND
C   DYP-LOOP                                                             GEND
       K      = 1                                                        GEND
       KS     = 1                                                        GEND
       KB     = 0                                                        GEND
       NBXR   = NCARAY(K)                                                GEND
       J1     = NTO-NTY+1                                                GEND
       J2     = NTO                                                      GEND
       SGR    = SG(KS)                                                   GEND
       CGR    = CG(KS)                                                   GEND
       DO  270  I=I1,I2                                                  GEND
       CALL        DYPZ(KB,KS,LS,IZ,I,J1,J2,NYFLAG,FLND,FLNE,SGR,CGR,REFC,GEND
     1   FMACH,KR,ARB,NBEA,LBO,LSO,JBO,DT,DTA                            GEND
     2 , YB, ZB, RIA, X, YS, ZS, DELX )                                  GEND
       WRITE  (ITP3)  (DT(J), J=J1,J2),(DTA(J), J=J1,J2)                 GEND
       IF  (I.EQ.NBARAY(K))  K=K+1                                       GEND
       IF  (I.EQ.NBXR)  GO TO  260                                       GEND
       GO TO  270                                                        GEND
   260 CONTINUE                                                          GEND
       KS     = KS+1                                                     GEND
       NBXR   = NBXR+NCARAY(K)                                           GEND
       SGR    = SG(KS)                                                   GEND
       CGR    = CG(KS)                                                   GEND
   270 CONTINUE                                                          GEND
       NYFLAG = 0                                                        GEND
       IZ     = 0                                                        GEND
```

```
      IF (NTZ.EQ.0)  GO TO  310                                          GEND 258
                                                                         GEND 259
C  DYZ-LOOP     **     ALSO USED FOR GENERATING THE  DYY  MATRIX  --  SEE GEND 260
C  COMMENT IN  DYY-LOOP  BELOW                                           GEND 261
      I1     = I2+1                                                      GEND 262
      I2     = I2+NTZ                                                    GEND 263
      KS     = NSTRIP+1                                                  GEND 264
      KB     = 1                                                         GEND 265
      SGR    = 0.0                                                       GEND 266
      CGR    = 1.0                                                       GEND 267
  280 CONTINUE                                                           GEND 268
      DO  300  I=I1,I2                                                   GEND 269
      LS     = LSO                                                       GEND 270
      IZ     = IZ+1                                                      GEND 271
      CALL        DYPZ(KB,KS,LS,IZ,I,J1,J2,NYFLAG,FLND,FLNE,SGR,CGR,REFC,GEND 272
     1   FMACH,KR,ARB,NBEA,LBC,LSO,JBO,DT,DTA                            GEND 273
     2 , YB, ZB, RIA, X, YS, ZS, DELX )                                  GEND 274
      WRITE  (ITP3)  (DT(J), J=J1,J2),(DTA(J), J=J1,J2)                  GEND 275
      IF (IZ.EQ.NBEA(1,KB)) GO TO 290                                    GEND 276
      GO TO  300                                                         GEND 277
  290 CONTINUE                                                           GEND 278
      IZ     = 0                                                         GEND 279
      KB     = KB+1                                                      GEND 280
      KS     = KS+1                                                      GEND 281
  300 CONTINUE                                                           GEND 282
  310 CONTINUE                                                           GEND 283
      IF (NYFLAG.NE.0)  GO TO 320                                        GEND 284
                                                                         GEND 285
C  DYY-LOOP     **     THIS LOOP IS REDUCED TO SETTING THE CORRECT INDICESGEND 286
C  AND USING THE  DYZ-LOOP  ABOVE                                        GEND 287
      I1     = NTO-NTY+1                                                 GEND 288
      I2     = NTO                                                       GEND 289
      NYFLAG = 1                                                         GEND 290
      KB     = LBC                                                       GEND 291
      KS     = LSO                                                       GEND 292
      SGR    =-1.0                                                       GEND 293
      CGR    = 0.0                                                       GEND 294
      GO TO  280                                                         GEND 295
  320 CONTINUE                                                           GEND 296
      REWIND   ITP1                                                      GEND 297
      REWIND   ITP2                                                      GE.D 298
      REWIND   ITP3                                                      GEND 299
      REWIND   ITP4                                                      GEND 300
      REWIND   ITP8                                                      GEND 301
      I1     = 1                                                         GEND 302
      I2     = NTP+NTZ                                                   GEND 303
      NYFLAG = 0                                                         GEND 304
      ITAPE  = 1                                                         GEND 305
  330 CONTINUE                                                           GEND 306
      DO 360  I=I1,I2                                                    GEND 307
      J1     = 1                                                         GEND 308
      J2     = NTP                                                       GEND 309
```

```
      READ    (ITAPE)  (DT(J), J=J1,J2),(DTA(J), J=J1,J2)                    GEND 310
      IF  (NTZ.EQ.0)   GO TO 340                                             GEND 311
      J1    = J2+1                                                           GEND 312
      J2    = J2+NTZ                                                         GEND 313
      READ    (ITP2)   (DT(J), J=J1,J2),(DTA(J), J=J1,J2)                    GEND 314
  340 CONTINUE                                                               GEND 315
      IF  (NTY.EQ.0)   GO TO 350                                             GEND 316
      J1    = J2+1                                                           GEND 317
      J2    = J2+NTY                                                         GEND 318
      READ    (ITP3)   (DT(J), J=J1,J2),(DTA(J), J=J1,J2)                    GEND 319
  350 CONTINUE                                                               GEND 320
      WRITE   (ITP8)  (DT(J), J=1,J2), (DTA(J), J=1,J2)                      GEND 321
      IF   (IOPT .EQ. 1)  WRITE (MASTDT)  (DT(J), J = 1, J2),                GEND 322
     1                                    (DTA(J), J = 1, J2)                GEND 323
      IF  (NPRINT.NE.3)   GO TO 360                                          GEND 324
      WRITE   (6,20)   I,J2                                                  GEND 325
      WRITE   (6,30)   (DT(J),   J=1 ,J2)                                    GEND 326
      WRITE   (6,25)   I,J2                                                  GEND 327
      WRITE   (6,30)   (DTA(J), J=1 ,J2)                                     GEND 328
      IF   ((J2/5) .LT. 25)  GO TO  360                                      GEND 329
      WRITE   (6, 10)                                                        GEND 330
  360 CONTINUE                                                               GEND 331
      IF  (NTY.EQ.0)   GO TO 370                                             GEND 332
      IF  (NYFLAG.NE.0)   GO TO 370                                          GEND 333
      NYFLAG = 1                                                             GEND 334
      I1    = I2+1                                                           GEND 335
      I2    = I2+NTY                                                         GEND 336
      ITAPE   = 4                                                            GEND 337
      GO TO 330                                                              GEND 338
  370 CONTINUE                                                               GEND 339
      RETURN                                                                 GEND 340
      END                                                                    GEND 341
```

```
      SUBROUTINE GENF(NMODES, NTOT  , NTP,NB, YB    , NSBETO, IPRINT,   GENF
     1             LINES, NTP6, NUTL1 , NUTL2 , NTPH   , NTPH4 , NTFORC,  GENF
     2             NTGF, MASTCP, MASTFZ, MASTFY, JKR, H, FMULT, NASYM,    GENF
     3             DELA, XIS1, XIS2, NSBEA, FORCE, GF, WORK, NG)          GENF
C                                                                         GENF
      DIMENSION NMODES(6), YB(1)                                          GENF
      DIMENSION H(1), FMULT(1), DELA(1), XIS1(1), XIS2(1), NSBEA(1)       GENF
      COMPLEX    FORCE(  1  ), GF(  1  ), WORK(  1  )                     GENF
      DIMENSION CODE(12)                                                  GENF
C                                                                         GENF
      DATA        IZERO, IREAD, IWRITE / 0, 1, 0/, GLCJ, CNOT / 2*1.0 /   GENF
      DATA        CODE / 4HDPUS, 4HFGPS, 4HDZOS, 4HFGZS, 4HDYOS, 4HFGYS,  GENF
     1                   4HDPOA, 4HFGPA, 4HDZOA, 4HFGZA, 4HDYOA, 4HFGYA / GENF
C                                                                         GENF
   10 FORMAT   ( 1H1 /// 10X, 22H***    SEGMENT  7    ***  ///  5X,      GENF
     1            68HCALCULATE FORCES FOR MODES AND GUST,  AND GENERALIZED GENF
     2FORCES              // )                                            GENF
   30 FORMAT   ( // 20H    FORCE COLUMN      ,I3, 13H OF MATRIX     , 1A4/) GENF
   40 FORMAT   ( // 20H    GUST FORCE COLUMN,I3, 13H OF MATRIX     , 1A4/) GENF
   50 FORMAT   ( // 32H    GENERALIZED FORCE COLUMN        , I3,          GENF
     1            13H OF MATRIX      , 1A4 / )                            GENF
   60 FORMAT   ( // 32H    GENERALIZED GUST FORCE COLUMN , I3,            GENF
     1            13H OF MATRIX      , 1A4 / )                            GENF
   70 FORMAT   ( 6F16.6 )                                                 GENF
   72 FORMAT   ( 1H1 /// )                                                GENF
      WRITE    (NTP6,10)                                                  GENF
C                                                                         GENF
      REWIND   NUTL1                                                      GENF
      REWIND   NUTL2                                                      GENF
      REWIND   NTGF                                                       GENF
      ISYM        =  1                                                    GENF
      NMODE = NMODES(1) + NG                                              GENF
      IZY         =  0                                                    GENF
      IF   (NB .EQ. 0)  GO TO  110                                        GENF
      IZY         =  1                                                    GENF
      NUTL        =  NUTL1                                                GENF
      REWIND   NTPH                                                       GENF
   80 CONTINUE                                                            GENF
      NSLEND      =  NSBETO + NB                                          GENF
      READ     (NTPH)  NMSB, NMASB                                        GENF
      WRITE    (NUTL )  NMSB, NMASB                                       GENF
      NMTO        =  NMSB                                                 GENF
      DO  104  L = 1, 2                                                   GENF
      DO  102  NM = 1, NMTO                                               GENF
      READ     (NTPH)  ID                                                 GENF
      READ(NTPH) (FMULT(I),I=1,NSLEND)                                    GENF
      LJ2         =  0                                                    GENF
      IH          =  0                                                    GENF
      DO  100  LB = 1, NB                                                 GENF
      LJ1         =  LJ2 + 1                                              GENF
      LJ2         =  LJ2 + NSBEA(LB)                                      GENF
      DO  90  LJ = LJ1, LJ2                                               GENF
```

```
      IH        =  IH + 1                                               GENF
      H(IH) = GLCJ*(FMULT(LJ)  + FMULT(LJ+1))  / 2.0                    GENF
   90 CONTINUE                                                          GENF
      LJ2       =  LJ2 + 1                                              GENF
  100 CONTINUE                                                          GENF
C                                                                       GENF
      WRITE  (NUTL )   ID                                               GENF
      WRITE  (NUTL )   (H(IH),   IH = 1, NSBETO)                        GENF
  102 CONTINUE                                                          GENF
      READ   (NTPH )   ID                                               GENF
      WRITE  (NUTL )   ID                                               GENF
      IF  (NMASB .EQ. 0)  GO TO  106                                    GENF
      NMTO      =  NMASB                                                GENF
  104 CONTINUE                                                          GENF
  106 CONTINUE                                                          GENF
      IF  (IZY .EQ. 2)  GO TO  110                                      GENF
      IZY       =  2                                                    GENF
      NUTL      =  NUTL2                                                GENF
      GO TO  80                                                         GENF
  110 CONTINUE                                                          GENF
C                                                                       GENF
      IF (JKR.GT.1) GO TO 500                                           GENF
C                                                                       GENF
C         WRITE THE TWO MATRICES  HPS  AND  HPA  ON TAPE  NTFORC  IN    GENF
C         COLUMN ORDER  PRECEDED BY  ROW- AND COLUMN-DIMENSIONS         GENF
C                                                                       GENF
      REWIND  NTPH4                                                     GENF
      IGO       =  1                                                    GENF
      READ   (NTPH4)  NSYM, NASYM                                       GENF
      NMODE     =  NSYM                                                 GENF
  330 CONTINUE                                                          GENF
      DO  340   J = 1, NMODE                                            GENF
      READ   (NTPH4)                                                    GENF
      READ   (NTPH4)  (H(I),   I = 1, NTP)                              GENF
      WRITE  (NTFORC) (H(I),   I = 1, NTP)                              GENF
  340 CONTINUE                                                          GENF
C                                                                       GENF
      IF  (NASYM .EQ. 0 .OR. IGO .EQ. 2)  GO TO  350                    GENF
      IGO       =  2                                                    GENF
      NMODE     =  NASYM                                                GENF
      READ   (NTPH4)                                                    GENF
      GO TO  330                                                        GENF
C                                                                       GENF
  350 CONTINUE                                                          GENF
      IF  (NB .EQ. 0)  GO TO  410                                       GENF
C                                                                       GENF
C         WRITE THE FOUR MATRICES  HZS,  HZA,  HYS,  HYA  ON TAPE       GENF
C         NTFORC  IN COLUMN ORDER PRECEDED BY ROW- AND COLUMN-DIMENSIONSGENF
C                                                                       GENF
      NUTL      =  NUTL1                                                GENF
      IGO       =  3                                                    GENF
  360 NMODE     =  NSYM                                                 GENF
```

```
      REWIND  NUTL
  370 READ    (NUTL)
C
      DO  390  J = 1, NMODE
      READ    (NUTL)
      READ    (NUTL)    (H(I),   I = 1, NSBETO)
      WRITE   (NTFORC) (H(I),   I = 1, NSBETO)
  390 CONTINUE
C
      IF  (IGO .EQ. 6)  GO TO  410
      IF  (NASYM .EQ. 0 .OR. IGO .EQ. 4)  GO TO  400
      NMODE     =  NASYM
      IGO       =  IGO + 1
      GO TO  370
  400 CONTINUE
      NUTL      =  NUTL2
      IGO       =  5
      IF  (NASYM .EQ. 0)  IGO = 6
      GO TO  360
C
  410 CONTINUE
C
C         H  MATRICES ARE NOW SAVED ON TAPE  NTFORC
C
  500 CONTINUE
C
      MM        =  0
      K2        =  0
      IGO       =  1
      NMODE     =  NMODES(1) + NG
      NMODE = NMODES(1) + NG
  122 NROW      =  NTP
      NTAPE     =  MASTCP
      NTPHS     =  NTPH4
      DO  130  I = 1, NROW
      FMULT(I) =  DELA(I)
  130 CONTINUE
      LENGTH    =  NTOT
  140 MM        =  MM + 1
      MODE      =  NMODES(MM)
      IF  (ISYM .EQ. 1)  REWIND  NTAPE
      K1        =  K2 + 1
      K2        =  K1 + 1
      MP        =  0
      IGUST     =  0
      MG        =  0
      NCOL      =  NMODE
C
      WRITE  (NTGF )  MODE, NCOL
C
      IBEG      =  -NROW + 1
C
```

```
C         READ INTO CORE  (WORK-ARRAY)  THE ENTIRE SYMMETRIC,  OR
C         ANTISYMMETRIC SOLUTION MATRIX (PRESSURES OR BODY FORCES)
C
      DO  150  N = 1, NMODE
      IBEG     =  IBEG + NROW
      CALL        RWREC(IREAD, NTAPE, WORK(IBEG), LENGTH, 1, 0 )
  150 CONTINUE
  160 CONTINUE
      MP       =  MP + 1
      KJ       =  (MP-1) * NROW
      DO  170  I = 1, NROW
      KJ       =  KJ + 1
      FORCE(I) =  FMULT(I) * WORK(KJ)
C
  170 CONTINUE
C         ONE COLUMN OF FORCES IS COMPLETE,  WRITE IT ON TAPE  NTFORC
C
      CALL        RWREC(IWRITE, NTFORC, FORCE, NROW, 1, 0 )
      IF  (IPRINT .EQ. 0)  GO TO  200
      IF  (IGUST  .NE. 0)  GO TO  180
      WRITE  (NTP6,30)  MP, CODE(K1)
      GO TO  190
  180 MG       =  MG + 1
      WRITE  (NTP6,40)  MG, CODE(K2)
  190 CONTINUE
      WRITE  (NTP6,70)  (FORCE(I),  I = 1, NROW)
  200 CONTINUE
C
C         BEGIN  GENERALIZED FORCES
C
      CALL        ZEROUT ( GF , NTOT,  1,  0 )
C
      REWIND  NTPHS
      READ   (NTPHS)  MODEH,  IDUMY
      IF   (ISYM .EQ. 1)  GO TO  208
      DO  204  NH = 1, MODEH
      READ   (NTPHS)
      READ   (NTPHS)
  204 CONTINUE
      READ   (NTPHS)          KODE
  208 CONTINUE
      DO  220  MD = 1, MODE
      READ   (NTPHS)  ICOL
      READ (NTPHS) (H(JH),JH=1,NROW)
      DO  210  IR = 1, NROW
      GF(MD)   =  GF(MD) + FORCE(IR) * H(IR)
  210 CONTINUE
  220 CONTINUE
C
C         WRITE  GENERALIZED FORCE COLUMN ON SCRATCH TAPE  NTGF
C
      CALL        RWREC(IWRITE, NTGF  ,  GF  ,  MODE,  1, 0 )
```

```
C
      IF  (MP .EQ. (NMODE-NG)) IGUST = 1
      IF  (MP .LT. NMODE)  GO TO  160
C
      GO TO  (230,250,260),  IGO
  230 CONTINUE
      IF  (NB .NE. 0)  GO TO  232
      MM        =  3
      K2        =  6
      GO TO  260
  232 CONTINUE
      NTAPE     =  MASTFZ
      NTPHS     =  NUTL1
      NROW      =  NSBETO
      LENGTH    =  NROW
      IGO       =  2
C
      JB        =  1
      JE        =  NSBEA(1)
      DO  240  I = 1, NROW
      FMULT(I) = 1.0
      IF  (ABS(YB(JB)) .LT. 0.0001)  FMULT(I) = 0.5
      IF  (I .LT. JE .OR. I .EQ. NROW)  GO TO  240
      JB        =  JB + 1
      JE        =  JE + NSBEA(JB)
  240 CONTINUE
      GO TO  140
  250 CONTINUE
      NTAPE     =  MASTFY
      NTPHS     =  NUTL2
      IGO       =  3
      GO TO  140
  260 CONTINUE
      IF  (ISYM .EQ. 2 .OR. NASYM .EQ. 0)  GO TO  270
      NMODE = NMODES(4) + NG
      ISYM      =  2
      IGO       =  1
      GO TO  122
C
  270 CONTINUE
C
C         READ BACK THE GENERALIZED FORCES FROM THE SCRATCH TAPE NTGF
C         INTO CORE  (WORK-ARRAY), AND WRITE ENTIRE MATRIX ON THE
C         MASTER TAPE  NTFORC ---' ALSO,  PRINT THE  GF  COLUMNS
C
      REWIND  NTGF
      K2        =  0
      I1        =  1
      IGO       =  1
C
  280 CONTINUE
      MG        =  0
```

```
      IGUST    =  0                                                     GENF 262
      READ   (NTGF )  NROW, NCOL                                        GENF 263
      K1       =  K2 + 1                                                GENF 264
      K2       =  K1 + 1                                                GENF 265
C                                                                       GENF 266
      ILAST = 0                                                         GENF 267
      DO  310  N = 1, NCOL                                              GENF 268
      IBEG  = ILAST+1                                                   GENF 269
      ILAST = ILAST+NROW                                                GENF 270
      CALL        RWREC(IREAD, NTGF , WORK(IBEG), NROW, 1, 0  )         GENF 271
      IF  (IGUST .NE. 0)  GO TO  290                                    GENF 272
      WRITE  (NTP6,50)  N , CODE(K1)                                    GENF 273
      GO TO  300                                                        GENF 274
  290 MG       =  MG + 1                                                GENF 275
      WRITE  (NTP6,60)  MG, CODE(K2)                                    GENF 276
  300 CONTINUE                                                          GENF 277
      WRITE  (NTP6,70)  (WORK(II), II = IBEG, ILAST )                   GENF 278
      IF  (N .EQ. (NCOL-NG)) IGUST = 1                                  GENF 279
  310 CONTINUE                                                          GENF 280
C                                                                       GENF 281
      CALL        RWREC(IWRITE, NTFORC, WORK(I1),  ILAST, 1, 0  )       GENF 282
C                                                                       GENF 283
      IF  (NB .EQ. 0 .AND. NASYM .EQ. 0)  GO TO  320                    GENF 284
      IF  (IGO.EQ. 3 .AND. NASYM .EQ. 0)  GO TO  320                    GENF 285
      IF  (IGO .EQ. 6)  GO TO  320                                      GENF 286
      WRITE  (NTP6, 72)                                                 GENF 287
      IGO      =  IGO + 1                                               GENF 288
      IF (NB.NE.0) GO TO 280                                            GENF 289
C                                                                       GENF 290
      IF  (IGO .GT. 2 .OR. NASYM .EQ. 0)  GO TO  320                    GENF 291
      IGO      =  4                                                     GENF 292
      K2       =  6                                                     GENF 293
      GO TO  280                                                        GENF 294
C                                                                       GENF 295
  320 RETURN                                                            GENF 296
      END                                                               GENF 297
```

```
      SUBROUTINE GEOM (DELA, IFLA, NBFA, NT12, NAS, NASB, ISSTR, NSSTR  GEOM     2
     1 , NSBEA, NBARAY, NCARAY, NSARAY, AO, YB, ZB, ARB, AVR, AOP, RIA   GEOM     3
     2 , XLE, XTE, THIA, XIS1, XIS2, X, CG, CS, EE, SG, YS, ZS, XIC, XIJGEOM     4
     3 , YIN, ZIN, DELX, XLAM, COORD, NOUT                               GEOM     5
     4 , TH, TAU, XII, RI, XIS, RS                                       GEOM     6
     5 , GMA, DYS, DZS, GMAR                                             GEOM     7
     * , XI1, XI2, ETA1, ETA2, ZETA1, ZETA2, XC                          GEOM     8
     7 , ETA, ZETA, ETAS, ZETAS, IHD, TEMP, ITEMP)                       GEOM     9
      DIMENSION DELA(1)                                                  GEOM    10
      DIMENSION IFLA(2,1), NBEA(2,1), NT12(2,1)                          GEOM    11
      DIMENSION NAS(1), NASB(1), ISSTR(1), NSSTR(1), NSBEA(1), NBARAY(1)GEOM    12
      DIMENSION NCARAY(1), NSARAY(1), AO(1), YB(1), ZB(1), ARB(1),AVR(1)GECM    13
      DIMENSION AOP(1), RIA(1), XLE(1), XTE(1), THIA(1), XIS1(1),XIS2(1)GFOM    14
      DIMENSION X(1), CG(1), CS(1), EE(1), SG(1), YS(1), ZS(1), XIC(1)   GEOM    15
      DIMENSION XIJ(1), YIN(1), ZIN(1), DELX(1), XLAM(1), COORD(1)       GEOM    16
      DIMENSION TH(1), TAU(1), XII(1), RI(1), XIS(1), RS(1)              GEOM    17
      DIMENSION GMA(1), DYS(1), DZS(1), GMAR(1)                          GEOM    18
      DIMENSION XI1(1), XI2(1), ETA1(1), ETA2(2), ZETA1(1), ZETA2(1)     GEOM    19
      DIMENSION XC(4,350)                                                GEOM    20
      DIMENSICN ETA(1), ZETA(1), ETAS(1), ZETAS(1)                       GEUM    21
      DIMENSION IHD(1),TEMP(1) , ITEMP(1)                                GEOM    22
C   ***   CALCULATES ARRAYS OF GEOMETRY FOR ALL LIFTING SURFACES         GEOM    23
C          AND ALL INTERFERENCE-  AND SLENDER BODIES                     GEOM    24
      COMMON /AROCOM/ NTI, MODES                                         AROCOM 2
     X , NP, MSTRIP, NSMAX, NCMAX, NTOTAL, NB, MSBE, MBE                 AROCOM 3
     Y , ND, NF, NBY, NBZ, NTO, NTP, NTY, NTZ                            AROCOM 4
     1 , NTYS, NTZS, MAXGR, MAXSTR, NSBETO, NSTRIP, KR, XM, REFA, REFC   AROCOM 5
     2 , REFS, FMACH, LINES                                              AROCOM 6
      DIMENSION TH1(20)                                                  GEOM    26
      REAL      KR                                                       GEOM    27
   10 FORMAT (6F12.0)                                                    GEOM    28
   20 FORMAT (6I12)                                                      GFOM    29
   30 FORMAT  (1H0,10X,I4,31H  CHORDWISE DIVISIONS FOR PANEL, I3 /)      GEOM    30
   50 FORMAT  (1H0,10X,I4,31H  SPANWISE  DIVISIONS FOR PANEL, I3 /)      GEOM    31
   60 FORMAT  ( 6E16.6 )                                                 GEOM    32
   70 FORMAT  (1H0/// 10X,13H**  PANEL NO.,I4,17H INPUT VALUES  **/ )    GEOM    33
   90 FORMAT (1H0,5X,4HX1 =,F12.6,6H  X2 =,F12.6, 6H  Y1 =,F12.6, 6H  Z1GEOM    34
     1 =,F12.6//6X,4HX3 =,F12.6,6H  X4 =,F12.6,6H  Y2 =,F12.6,6H  Z2 =, GEOM    35
     2          F12.6//6X,4HNC =,I3,6H  NS =,I3//                        GEOM    36
     3                  6X, 19HGROUP NUMBER        = , I5  //  )         GEOM    37
  100 FORMAT (1H1//25X,12H**  BODY NO.,I4,18H  INPUT VALUES  **// 15X,   GEOM    38
     1 32HCENTER OF BODY COORDINATES    Y =, F13.6,5X, 3HZ =, F13.6/ 15X,GEOM    39
     2 30HAVERAGE HALF-WIDTH OF BODY    =, F15.6/ 15X, 30HCROSS-SECTIONALGEOM    40
     3 ASPECT RATIO =,F15.6/ 15X, 41HNUMBER OF INTERFERENCE ELEMENTS ON GEOM    41
     4BODY =, I4/ 15X,33HNUMBER OF SLENDER BODY ELEMENTS =,I12/15X,24HZ-YGEOM    42
     5 ORIENTATION FLAG    =,I3/15X, 9HRI-FLAG =, I3,12H  R-S FLAG =,I3/)GEOM    43
  110 FORMAT  (1H0/// 10X,29H**  SUMMARY OF PANEL DATA  **// 7X,50HPANELGEOM    44
     1    NC     NS  NB-ARRAY  DIHEDRAL                 /               GEOM    45
     2             36X, 18HANGLE                /)                        GEOM    46
  120 FORMAT  ( 1H1 /// 10X, 22H***   SEGMENT  1   ***  ///  5X,         GEOM    47
     *             25HGENERATE GEOMETRY ARRAYS   //  )                   GEOM    48
  130 FORMAT  (1H0,10X,I4,24H  XI-I ELEMENTS FOR BODY, I3 /)             GEOM    49
```

```
  140 FORMAT  (1H0,10X,I4,24H  R-I  ELEMENTS FOR BODY, I3 /)             GEOM  50
  150 FORMAT  (1H0,6X,3I5,I8,F12.5,7X,I3 )                                GEOM  51
  160 FORMAT  (1H1///30X,38H**  GEOMETRY ARRAYS FOR ALL PANELS  **//5X,   GEOM  52
     157HPANEL  STRIP  BOX    3/4 CHORD      1/4 CHORD X-COORDINATES, 7X, GEOM  53
     211H  BOX CHORD,5X, 9H1/4 CHORD/ 7X,15HNO.   NO.   NO.,7X,1HX,7X,    GECM  54
     359HINBOARD     CENTER     OUTBOARD     DELTA-X      SWEEP ANGLE /)  GEOM  55
  170 FORMAT  (1H0,10X,I4,24H  XI-S ELEMENTS FOR BODY, I3 /)             GEOM  56
  180 FORMAT  (1H0,10X,I4,24H  R-S  ELEMENTS FOR BODY, I3 /)             GEOM  57
  190 FORMAT  (     4X,3I5,6F12.5)                                        GEOM  58
  200 FORMAT  (1H0,10X,I4,27H  THETA   ELEMENTS FOR BODY, I3 /)          GEOM  59
  202 FORMAT  ( 1H1 /// 10X, 3HBOX, 6X, 3HXC1, 10X, 3HXC2, 10X, 3HXC3     GEOM  60
     1                  ,  10X,  3HXC4       /   )                        GEOM  61
  210 FORMAT  (1H1//8X,6HSTRIP     ,7X, 1HY,11X,1HZ,8X,19HDELTA-Y     DELGEOM  62
     1TA-Z,7X, 1HE, 10X,18HCHORD         X-L.E. /)                        GFOM  63
  220 FORMAT  (1H1/// 30X, 38H**  GEOMETRY ARRAYS FOR ALL BODIES  ** //  GEOM  64
     *            10X, 12HINTERFERENCE  / 7X, 15HBODY    SEGMENT , 10X,   GEOM  65
     *            3HX-I, 10X, 3HY-I, 13X, 3HZ-I,13X,3HR-I,11X,5HGAMMA / 6X, GEOM  66
     *            16HNUMBER      NUMBER   /   )                           GEOM  67
  230 FORMAT  (1H1/// 10X, 7HSLENDER  / 7X, 15HBODY    SEGMENT , 9X,      GEOM  68
     *            4HXIS1,11X, 4HXIS2, 13X, 2HA0 ,10X, 9HA0-PRIME / 6X,    GEOM  69
     *            16HNUMBER      NUMBER   /   )                           GEOM  70
  235 FORMAT  ( 2I10, 5F16.5 )                                            GEOM  71
  250 FORMAT  (      10X,I3,7F12.5)                                       GEOM  72
  255 FORMAT  ( 1H1 /// )                                                 GFOM  73
C                                                                         GEOM  74
      WRITE  (6,120)                                                      GEOM  75
      PI = 3.1415926                                                      GEOM  76
      NBTO = 0                                                            GEOM  77
      NSTC = 0                                                            GEOM  78
      K    = 0                                                            GFOM  79
      JS   = 0                                                            GEOM  80
      LAS  = 0                                                            GEOM  81
      NZY  = 2                                                            GEOM  82
      LP1     = 1                                                         GECM  83
      LP2     = NP                                                        GEOM  84
      IOLD       =   0                                                    GEOM  85
      MAXSTR     =   0                                                    GFOM  86
      MAXGR      =   0                                                    GECM  87
      NSTRIP     =   0                                                    GEOM  88
      DO  350  LP = LP1, LP2                                              GFCM  89
      WRITE (6,70)  LP                                                    GEOM  90
      J     = 0                                                           GEOM  91
C                                                                         GEOM  92
C  **   READ PANEL COORDINATES, NUMBER OF CHORDWISE BOXES AND (SPANWISE)GEOM  93
C       STRIPS,  AND THE ASSOCIATED BODIES                                GEOM  94
C                                                                         GEOM  95
      READ (NTI,10) X1, X2, X3, X4                                        GEOM  96
      READ (NTI,10) Y1, Y2, Z1, Z2                                        GEOM  97
      READ (NTI,20) NC, NS, IGRUP                                         GEOM  98
      IF  (IGRUP .EQ. 0)  IGRUP = 1                                       GEOM  99
C                                                                         GEOM 100
C                                                                         GEOM 101
```

```
      WRITE (6,90)  X1,X2,Y1,Z1, X3,X4,Y2,Z2, NC,NS,                    GEOM 102
     1                                        IGRUP                     GEOM 103
      NSTRIP   =  NSTRIP + NS                                           GEOM 104
      IF  (IGRUP .GT. MAXGR)  MAXGR = IGRUP                             GEOM 105
C                                                                       GEOM 106
C  **   STORE  Y- AND Z-COORDINATE OF INBOARD EDGE OF PANEL IN THE      GEOM 107
C       ARRAYS   'YIN'  ,  AND   'ZIN'    RESPECTIVELY                  GEOM 108
C                                                                       GEOM 109
      YIN(LP) = Y1                                                      GEOM 110
      ZIN(LP) = Z1                                                      GEOM 111
      NAT     =  NB                                                     GEOM 112
      IF  (NB.EQ.0)  GO TO  32C                                         GEOM 113
      DO  310  NA = 1, NB                                               GEOM 114
      LAS  = LAS+1                                                      GEOM 115
      NASB(LAS)=  NA                                                    GEOM 116
C  **   'NASB'     IS THE ARRAY OF ASSOCIATED BODIES FOR ALL PANELS     GEOM 117
  310 CONTINUE                                                          GEOM 118
      NAS(LP) = NAT                                                     GEOM 119
C  **   'NAS'     IS THE ARRAY OF THE NUMBER OF ASSOCIATED BODIES PER   GEOM 120
C  PANEL  --  THE MAXIMUM NUMBER OF ASSOCIATED BODIES PER PANEL IS  10  GEOM 121
  320 CONTINUE                                                          GEOM 122
      NCNS = NC*NS                                                      GEOM 123
      NCARAY(LP) = NC                                                   GEOM 124
      NSARAY(LP) = NS                                                   GEOM 125
      NBARAY(LP) = NBTO+NCNS                                            GEOM 126
      NSTO = NSTO+NS                                                    GEOM 127
      NCP1 = NC+1                                                       GEOM 128
      NSP1 = NS+1                                                       GEOM 129
C                                                                       GEOM 130
C  **   READ THE FRACTIONAL CHORDWISE-  AND SPANWISE DIVISIONS FOR      GEOM 131
C       PANEL  --  ARRAYS    'TH'   AND   'TAU'  RESPECTIVELY           GEOM 132
C                                                                       GEOM 133
      READ (NTI,10) (TH(I), I=1,NCP1)                                   GEOM 134
      READ (NTI,10) (TAU(I), I=1,NSP1)                                  GEOM 135
      WRITE (6,30)  NCP1,LP                                             GEOM 136
      WRITE (6,60)  (TH(I),   I=1,NCP1)                                 GEOM 137
      WRITE (6,50)  NSP1,LP                                             GEOM 138
      WRITE (6,60)  (TAU(I),  I=1,NSP1)                                 GEOM 139
C                                                                       GEOM 140
C  **   COMPUTE DIHEDRAL ANGLE FOR PANEL  --  ARRAY NAME IS   'GMA'     GEOM 141
C                                                                       GEOM 142
      YDIF = Y2-Y1                                                      GEOM 143
      ZDIF = Z2-Z1                                                      GEOM 144
      CALL        ATAN3(ZDIF,YDIF,ATANA)                                GEOM 145
      GMA(LP) = ATANA*180.0/PI                                          GEOM 146
      FOUR = X4-X3-X2+X1                                                GEOM 147
      DIF31= X3-X1                                                      GEOM 148
      DIF21= X2-X1                                                      GEOM 149
      DIF43= X4-X3                                                      GEOM 150
      DO 340  J=1,NS                                                    GEOM 151
      JS   = JS+1                                                       GEOM 152
      JP1  = J +1                                                       GEOM 153
```

```
      FACT = 0.5*(TAU(J)+TAU(JP1))
      TDIF = TAU(JP1) - TAU(J)
      DO 330  I=1,NC
      IP1  =  I+1
      K    =  K+1
      TH13 = 0.25*TH(I) + 0.75*TH(IP1)
      TH31 = 0.75*TH(I) + 0.25*TH(IP1)
C
C  **   COMPUTE THE  3/4-  AND  1/4-CHORD X-COORDINATES OF ALL PANEL
C       BOXES  *  ALSO, THE AVERAGE BOX-CHORD  --  STORE THEM IN THE
C       ARRAYS  'X' ,  'XI1',  'XI2'  AND  'DELX'  RESPECTIVELY
C
      X(K)    = FACT    * (TH13*FOUR+DIF31) + TH13*DIF21 + X1
      XI1(K)  = TAU(J)  * (TH31*FOUR+DIF31) + TH31*DIF21 + X1
      XI2(K)  = TAU(JP1)* (TH31*FOUR+DIF31) + TH31*DIF21 + X1
      DELX(K)= FACT * (TH(IP1)-TH(I))*FOUR + (TH(IP1)-TH(I))*DIF21
      DELA(K)= TDIF * DELX(K) * SQRT(YDIF**2 + ZDIF**2)
C
      XC(1,K)= TAU(J)   * (TH(I)   *FOUR+DIF31) + TH(I)    * DIF21 + X1
      XC(2,K)= TAU(J)   * (TH(IP1)*FOUR+DIF31) + TH(IP1)  * DIF21 + X1
      XC(3,K)= TAU(JP1) * (TH(I)   *FOUR+DIF31) + TH(I)    * DIF21 + X1
      XC(4,K)= TAU(JP1) * (TH(IP1)*FOUR+DIF31) + TH(IP1)  * DIF21 + X1
C
      IF  (ABS(YDIF) .GT. 0.0001)  GO TO  330
      IF  (ABS(Y1) .LE. 0.0001)  DELA(K) = 0.5 * DELA(K)
  330 CONTINUE
C
C  **   COMPUTE DATA ARRAYS FOR ALL STRIPS OF PANEL  --
C        'YS'       =  Y-COORDINATE OF CENTERLINE OF STRIP
C        'ZS'       =  Z-COORDINATE OF CENTERLINE OF STRIP
C        'DYS'      =  DELTA-Y  OF STRIP
C        'DZS'      =  DELTA-Z  OF STRIP
C        'EE'       =  SEMI-WIDTH OF STRIP
C        'CS'       =  CHORDLENGTH OF STRIP
C        'SG'       =  SINE   OF THE DIHEDRAL ANGLE OF STRIP
C        'CG'       =  COSINE OF THE DIHEDRAL ANGLE OF STRIP
C        'XIJ'      =  X-COORDINATE OF LEADING EDGE OF STRIP
C        'GMAR'     =  DIHEDRAL ANGLE OF STRIP IN RADIANS
C
      YS(JS) = FACT * YDIF + Y1
      ZS(JS) = FACT * ZDIF + Z1
      DYS(JS)= TDIF * YDIF
      DZS(JS)= TDIF * ZDIF
      EE(JS) = 0.5*SQRT( DYS(JS)**2 + DZS(JS)**2)
      CS(JS) = FACT * FOUR + DIF21
      SG(JS) = DZS(JS)/(2.0*EE(JS))
      CG(JS) = DYS(JS)/(2.0*EE(JS))
      XIJ(JS)= FACT * DIF31 + X1
      GMAR(JS) = GMA(LP)*PI/180.0
      COORD(JS)=  YS(JS)
      IF  (ABS(CG(JS)) .LE. 0.0001)  COORD(JS) = ZS(JS)
  340 CONTINUE
```

```
      NBTC = NBARAY(LP)                                                 GEOM 206
C                                                                       GEOM 207
      CALL       GRUP(NS, JS, IOLD, IGRUP, MAXSTR, ISSTR, NSSTR, COORD) GEOM 208
C                                                                       GEOM 209
      WRITE  (6,255)                                                    GEOM 210
  350 CONTINUE                                                          GEOM 211
      NBOX = NBARAY(NP)                                                 GEOM 212
      IHD(7) = NBOX                                                     GEOM 213
      WRITE (6,110)                                                     GEOM 214
      I1   = 0                                                          GEOM 215
      I2   = 0                                                          GEOM 216
      NTP  = NBOX                                                       GEOM 217
C                                                                       GEOM 218
      DO 360  L = 1,NP                                                  GEOM 219
      I1   = I2+1                                                       GEOM 220
      I2   = I2+NAS(L)                                                  GEOM 221
C  DIHEDRAL ANGLE PER PANEL IS GIVEN IN DEGREES                         GEOM 222
      WRITE (6,150)  L,NCARAY(L),NSARAY(L),NBARAY(L),GMA(L)             GEOM 223
  360 CONTINUE                                                          GEOM 224
      WRITE (6,160)                                                     GEOM 225
      LP   = 1                                                          GEOM 226
      J    = 1                                                          GEOM 227
      NBXP = NBARAY(1)                                                  GEOM 228
      NBXS = NCARAY(1)                                                  GEOM 229
      SIGN = 1.0                                                        GEOM 230
      LPAGE    =  1                                                     GEOM 231
      DO 390  I=1,NBOX                                                  GEOM 232
      IF (I.LE.NBXP)  GO TO 370                                         GEOM 233
      LP   = LP+1                                                       GEOM 234
      NBXP = NBARAY(LP)                                                 GEOM 235
  370 CONTINUE                                                          GEOM 236
      IF (I.LE.NBXS)  GO TO 380                                         GEOM 237
      J    = J+1                                                        GEOM 238
      NBXS = NBXS+NCARAY(LP)                                            GEOM 239
  380 CONTINUE                                                          GEOM 240
      XIC(I)  =   0.5*(XI1(I)+XI2(I))                                   GEOM 241
      UP   = XI2(I)-XI1(I)                                              GEOM 242
      DOWN = 2.0*EE(J)*SIGN                                             GEOM 243
      CALL       ATAN3(UP,DOWN,ATANX)                                   GEOM 244
      XLAM(I) = ATANX                                                   GEOM 245
      WRITE (6,190)  LP,J,I,X(I),XI1(I),XIC(I), XI2(I), DELX(I),XLAM(I) GEOM 246
      ETA ( I) =  YS(J)                                                 GEOM 247
      ZETA( I) =  ZS(J)                                                 GEOM 248
      ETA1( I) =  YS(J) - .5 * DYS(J)                                   GEOM 249
      ZETA1(I) =  ZS(J) - .5 * DZS(J)                                   GEOM 250
      ETA2( I) =  YS(J) + .5 * DYS(J)                                   GEOM 251
      ZETA2(I) =  ZS(J) + .5 * DZS(J)                                   GEOM 252
C                                                                       GEOM 253
C  SWEEP ANGLE IS WRITTEN IN RADIANS,  BUT IT IS STORED IN  'XLAM'   ASGEOM 254
C  TANGENT LAMBDA                                                       GEOM 255
      XLAM(I) = UP/DOWN                                                 GEOM 256
      IF  (LPAGE*LINES .NE. I )  GO TO  390                             GEOM 257
```

```
      LPAGE    =  LPAGE + 1
      WRITE  (6,160)
  390 CONTINUE
C
      WRITE  (6,202)
      DO  392  I = 1, NBOX
      WRITE  (6,250)  I, (XC(J,I), J = 1, 4)
  392 CONTINUE
      WRITE (6,210)
      LPAGE    =  1
      DO 400  J=1,NSTRIP
      WRITE (6,250)  J,YS(J),ZS(J),DYS(J),DZS(J),EE(J),CS(J),XIJ(J)
      JL       =  ISSTR(J)
      COORD(JL)=  COORD(J)
      IF  (LPAGE*LINES .NE. J )  GO TO  400
      LPAGE    =  LPAGE + 1
      WRITE  (6,255)
      WRITE  (6,210)
  400 CONTINUE
      I2   = NBOX
      J    = NSTRIP
      NTO  = NBOX
      KI   = 0
      IS2  = 0
      LBN  = 0
      NBZ  = 0
      NBY  = 0
      NTZ  = 0
      NTY  = 0
      NTZS = 0
      NTYS = 0
      NSBETO   =  0
      IT12 = 0
C  **   PANEL GEOMETRY COMPUTATIONS COMPLETE --  BYPASS BODY  DATA-READ
C       AND ARRAY FORMATIONS  IF  NB=0
C
      IF (NB.EQ.0)  GO TO 560
      DO 550  LB=1,NB
C
C  **   READ BODY DATA
C
      READ (NTI,10) ZC, YC, RAD, AR
      READ (NTI,20) NBE, NSBE, NRI, NRS, NT1
      WRITE (6,100)  LB,YC,ZC,RAD,AR,NBE,NSBE,NZY,NRI,NRS
C  **  NZY=1  MEANS  BODY IS Z-ORIENTED
C  **  NZY=2  MEANS  BODY IS BOTH Z-  AND Y-ORIENTED
C  **  NZY=3  MEANS  BODY IS Y-ORIENTED
C  **  NOTE THAT BODY DATA HAS TO BE INPUT IN THIS ORDER,  I.E.
C  Z-ORIENTED BODIES FIRST,  Z&Y ORIENTED BODIES NEXT,  AND  Y-ORIENTED
C  BODIES LAST
      NTO  = NTO + NBE
      NSBETO   =  NSBETO + NSBE
```

```
      GO TO (410,410,420), NZY
  410 CONTINUE
      NBZ  =NBZ+1
      NTZ  = NTZ +NBE
      NTZS = NTZS+NSBE
      IF (NZY.EQ.1)  GO TO 430
  420 CONTINUE
      NBY  = NBY+1
      NTY  = NTY +NBE
      NTYS = NTYS+NSBE
  430 CONTINUE
      LBN  = LBN+1
C
C  **   SAVE BODY ARRAYS  --
C        ZB(10)       =  Z-COORDINATE OF BODY CENTER
C        YB(10)       =  Y-COORDINATE OF BODY CENTER
C        AVR(10)      =  AVERAGE SEMI-WIDTH OF BODY
C        ARB(10)      =  CROSS SECTIONAL ASPECT RATIO OF BODY
C        NBEA(10,1)   =  NUMBER OF INTERFERENCE BODY ELEMENTS
C        NBEA(10,2)   =  Z-Y (ORIENTATION) FLAG FOR BODY
C        NSBEA(10)    =  NUMBER OF SLENDER BODY ELEMENTS
C
      ZB( LBN) = ZC
      YB( LBN) = YC
      AVR(LBN) = RAD
      ARB(LBN) = AR
      NBEA(1,LBN) = NBE
      NBEA(2,LBN) = NZY
      NSBEA(LBN) = NSBE
      NBEP1= NBE+1
C
C  **   IF NBE = 0  (NO INTERFERENCE BODY ELEMENTS),  BYPASS READING OF
C       ALL INTERFERENCE BODY ARRAYS
C
      IF (NBE.EQ.0)  GO TO  47C
C
C  **   READ  X-COORDINATES OF INTERFERENCE BODY ELEMENT ENDPOINTS
C
      READ (NTI,10) (XII(I), I=1,NBEP1)
      WRITE (6,130)  NBEP1, LB
      WRITE (6,60)  (XII(I),   I=1,NBEP1)
      IF (NRI.NE.0)  GO TO 450
      DO 440  I=1,NBEP1
      RI(I)= RAD
  440 CONTINUE
      GO TO  460
  450 CONTINUE
C
C  **   READ  RADII  OF INTERFERENCE BODY ELEMENT ENDPOINTS  --  ONLY
C       IF  NRI = 1
C
      READ (NTI,10) (RI(I), I=1,NBEP1)
```

```
  460 CONTINUE                                                      GEOM 362
      WRITE (6,140)  NBEP1, LB                                      GEOM 363
      WRITE (6,60)   (RI(I),     I=1,NBEP1)                         GEOM 364
C  **   READ  THETA-1-MU  =  ANGULAR ORIENTATION OF POINT  MU  ON BODY  GEOM 365
C       --  INPUT  THETA-1-MU  IN DEGREES --                        GEOM 366
C       --  SAVE CONTINUOUS ARRAY FOR ALL BODIES IN THIA(100) IN RADIANSGEOM 367
C                                                                   GEOM 368
      READ (NTI,10) (TH1(I), I=1,NT1)                               GEOM 369
      WRITE (6,200)  NT1,    LB                                     GEOM 370
      WRITE (6,60)   (TH1(I),      I=1,NT1)                         GEOM 371
      IA   = 0                                                      GEOM 372
      IT11 =  IT12+1                                                GEOM 373
      IT12 =  IT12+NT1                                              GEOM 374
      DO 520 IT=IT11,IT12                                           GEOM 375
      IA   = IA+1                                                   GEOM 376
      THIA(IT) = TH1(IA)*PI/180.0                                   GEOM 377
  520 CONTINUE                                                      GEOM 378
  470 CONTINUE                                                      GEOM 379
      NSBEP1   =   NSBE + 1                                         GEOM 380
C  **   IF NSBE= 0  (NO SLENDER BODY ELEMENTS ),  BYPASS READING OF GEOM 381
C       ALL SLENDER BODY ARRAYS                                     GEOM 382
C                                                                   GEOM 383
      IF (NSBE.EQ.0) GO TO  510                                     GEOM 384
C                                                                   GEOM 385
C  **   READ  X-COORDINATES OF SLENDER BODY ELEMENT ENDPOINTS       GEOM 386
C                                                                   GEOM 387
      READ (NTI,10) (XIS(I), I=1,NSBEP1)                            GEOM 388
      WRITE (6,170)  NSBEP1,LB                                      GEOM 389
      WRITE (6,60)   (XIS(I),   I=1,NSBEP1)                         GEOM 390
      IF (NRS.NE.0)  GO TO 490                                      GEOM 391
      DO 480  I=1,NSBEP1                                            GEOM 392
      RS(I)= RAD                                                    GEOM 393
  480 CONTINUE                                                      GEOM 394
      GO TO  500                                                    GEOM 395
  490 CONTINUE                                                      GEOM 396
C                                                                   GEOM 397
C  **   READ  RADII  OF SLENDER BODY ELEMENT ENDPOINTS  --  ONLY    GEOM 398
C       IF  NRS = 1                                                 GEOM 399
C                                                                   GEOM 400
      READ (NTI,10) (RS(I), I=1,NSBEP1)                             GEOM 401
  500 CONTINUE                                                      GEOM 402
      WRITE (6,180)  NSBEP1,LB                                      GEOM 403
      WRITE (6,60)   (RS(I),   I=1,NSBEP1)                          GEOM 404
  510 CONTINUE                                                      GEOM 405
C                                                                   GEOM 406
      IFLA(1,LB) = 1                                                GEOM 407
      IFLA(2,LB) = NBE                                              GEOM 408
      NT12(1,LBN) = NT1                                             GEOM 409
      NT12(2,LBN) = NT2                                             GEOM 410
      I1   = I2+1                                                   GEOM 411
      I2   = I2+NBE                                                 GEOM 412
      K    = 0                                                      GEOM 413
```

```
C                                                                        GEOM 414
C   **    APPEND THE X-, XI1-, XI2-, DELX-, YS-, ZS- AND GMAR-ARRAYS  TO  GECM 415
C         END UP WITH CONTINUOUS ARRAYS FOR ALL RECEIVING POINTS          GEOM 416
C                                                                        GEOM 417
      DO 530  I=I1,I2                                                    GEOM 418
      K     = K+1                                                        GFOM 419
      KP1   = K+1                                                        GEOM 420
      KI    = KI+1                                                       GEOM 421
      IP1   = I+1                                                        GEOM 422
      X(I)       = 0.5*(XII(KP1)+XII(K))                                 GEOM 423
      XI1(I)     = XII(K)                                                GEOM 424
      XI2(I)     = XII(KP1)                                              GEOM 425
      DELX(I)    = XI2(I)-XI1(I)                                         GEOM 426
C                                                                        GEOM 427
C   **    COMPUTE RADII OF ALL INTERFERENCE BODY ELEMENT MIDPOINTS  --    GEOM 428
C         SAVE CONTINUOUS ARRAY FOR ALL BODIES IN  RIA(100)              GEOM 429
C                                                                        GEOM 430
      RIA(KI)    = 0.5*(RI(K) + RI(KP1))                                 GEOM 431
  530 CONTINUE                                                           GEOM 432
      J     = J+1                                                        GEOM 433
      YS(J)      = YC                                                    GEOM 434
      ZS(J)      = ZC                                                    GEOM 435
C                                                                        GEOM 436
C   **    SAVE LEADING EDGE-  AND TRAILING EDGE X-COORDINATES OF ALL      GEOM 437
C         INTERFERENECE BODIES IN THE ARRAYS  XLE(10)  AND  XTE(10)       GEOM 438
C         RESPECTIVELY                                                   GEOM 439
      XLE(LBN)   = XII(1)                                                GEOM 440
      XTE(LBN)   = XII(NBEP1)                                            GEOM 441
      CS(J)      = XTE(LBN)-XLE(LBN)                                     GEOM 442
      GMAR(J) = 0.0                                                      GEOM 443
      IF (NZY.EQ.3)   GMAR(J) =-PI/2.0                                   GEOM 444
      IS1   = IS2+1                                                      GEOM 445
      IS2   = IS2+NSBE                                                   GEOM 446
      KS    = 0                                                          GEOM 447
C                                                                        GEOM 448
C   **    SAVE THE FOLLOWING CONTINUOUS ARRAYS FOR ALL  SLENDER BODIES  --GEOM 449
C          'XIS1'    = X-COORDINATE OF LEADING EDGE OF SLENDER BODY ELEM.  GEOM 450
C          'XIS2'    = X-COORDINATE OF TRAILING EDGE OF SLENDER BODY ELEM. GEOM 451
C          'AO'      = AVERAGE RADIUS OF SLENDER BODY ELEMENT (Y-DIRECTION)GFOM 452
C          'AOP'     = DERIVATIVE OF THE LOCAL AVERAGE RADIUS  AO          GEOM 453
C                                                                        GEOM 454
      DO 540  IS=IS1,IS2                                                 GFOM 455
      KS    = KS+1                                                       GEOM 456
      KSP1 = KS+1                                                        GEOM 457
      XIS1(IS) = XIS(KS)                                                 GEOM 458
      XIS2(IS) = XIS(KSP1)                                               GEOM 459
      AO(IS)   = 0.5*(RS(KS)+RS(KSP1))                                   GEOM 460
      AOP(IS)  = (RS(KSP1)-RS(KS)) / (XIS2(IS)-XIS1(IS))                 GEOM 461
      ETAS (IS)=  YC                                                     GEOM 462
      ZETAS(IS)=  ZC                                                     GEOM 463
C                                                                        GFOM 464
  540 CONTINUE                                                           GEOM 465
```

```
  550 CONTINUE
  560 CONTINUE
C
      IHD(8) = NSBETO
       IHD(9) = NSTRIP
       IHD(10) = MAXSTR
      CALL WRAERO (4HHEAD,0,IHD,NOUT,NER)
      DO 561 I=1,NBOX
      TEMP(I) = DELA(I)
      TEMP(I+NBOX) = DELX(I)
  561 TEMP(I+2*NBOX) = XIC(I)
       N = 3*NBOX
       DO 562 I=1,NSTRIP
      N = N+1
      TEMP(N) = CG(I)
      TEMP(N+NSTRIP) = CS(I)
      TEMP(N+2*NSTRIP) = EE(I)
      TEMP(N+3*NSTRIP) = SG(I)
      TEMP(N+4*NSTRIP) = YS(I)
      TEMP(N+5*NSTRIP) = ZS(I)
      TEMP(N+6*NSTRIP) = XIJ(I)
  562 ITEMP(N+7*NSTRIP) = ISSTR(I)
      N = N+7*NSTRIP
      DO 563 I=1,MAXSTR
      N = N+1
  563 TEMP(N) = COORD(I)
      DO 564 I=1,NP
      N = N+1
      ITEMP(N) = NBARAY(I)
  564 ITEMP(N+NP) = NCARAY(I)
      N = N+NP
      DO 565 I=1,NB
      N = N+1
      ITEMP(N) = NSBEA(I)
      TEMP(N+NB) = YB(I)
  565 TEMP(N+2*NB) = ZB(I)
      CALL WRAERO (4HGEOM,0,TEMP,NOUT,NER)
      DO 567 I=1,NBOX
      TEMP(I) = XI1(I)
      TEMP(I+NBOX) = ETA1(I)
      TEMP(I+2*NBOX) = ZETA1(I)
      TEMP(I+3*NBOX) = XI2(I)
      TEMP(I+4*NBOX) = ETA2(I)
  567 TEMP(I+5*NBOX) = ZETA2(I)
      CALL WRAERO (4HGEO1,0,TEMP,NOUT,NER)
      DO 568 I=1,NBOX
      TEMP(I) = X(I)
      TEMP(I+NBOX) = ETA(I)
  568 TEMP(I+2*NBOX) = ZETA(I)
      CALL WRAERO (4HGEO3,0,TEMP,NOUT,NER)
C
      IF  (NB .EQ. 0)  GO TO  610
```

```
C
      DO 569 I=1,NSBETO
      TEMP(I) = XIS1(I)
      TEMP(I+NSBETO) = ETAS(I)
      TEMP(I+2*NSBETO) = ZETAS(I)
      TEMP(I+3*NSBETO) = XIS2(I)
      TEMP(I+4*NSBETO) = ETAS(I)
  569 TEMP(I+5*NSBETO) = ZETAS(I)
      CALL WRAERO (4HGEOS,0,TEMP,NOUT,NER)
C
      JB          =   NSTRIP
      IB          =   NBARAY(NP)
      WRITE  (6,220)
      LPAGE       =   1
      DO   580  LB = 1, NB
      JB          =   JB + 1
      I2 = NBEA(1,LB)
      DO   570  I = 1, I2
      IB          =   IB + 1
      KB          =   IB - NBOX
      WRITE  (6,235)  LB, KB, X(IB), YS(JB), ZS(JB), RIA(KB), GMAR(JB)
      IF   (LPAGE*LINES .NE. KB)  GO TO  570
      LPAGE       =   LPAGE + 1
      WRITE  (6,220)
  570 CONTINUE
  580 CONTINUE
C
      WRITE  (6,230)
      LPAGE       =   1
      IB          =   0
      DO   600  LB = 1, NB
      I2          =   NSBEA(LB)
      DO   590  I = 1, I2
      IB          =   IB + 1
      WRITE  (6,235)  LB, IB, XIS1(IB), XIS2(IB), AO(IB), AOP(IB)
      IF   (LPAGE*LINES .NE. IB)  GO TO  590
      LPAGE       =   LPAGE + 1
      WRITE  (6,230)
  590 CONTINUE
  600 CONTINUF
C
  610 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE GRUP(NS, J2, IOLD, IGRUP, MAXSTR, ISSTR, NSSTR, COORD)
C
      DIMENSION  NSSTR( 50), ISSTR(100), COORD(110)
   10 FORMAT  (//// 10X, *NS, J2, IOLD, IGRUP, MAXSTR * , 6I4 )
   20 FORMAT  ( /    10X, *ISSTR *, 5(5F16.8/) /)
   30 FORMAT  ( /    10X, *COORD *, 5(5F16.8/) /)
C
      JLAST      =   J2 - NS
      J1         =   JLAST + 1
      IF   (IOLD .NE. IGRUP)   GO TO   70
C
      DO   60   I = J1, J2
      DO    40   J = 1, JLAST
      IF   (COORD(I) .EQ. COORD(J))   GO TO   50
   40 CONTINUE
      MAXSTR     =   MAXSTR + 1
      NSSTR(IGRUP) = NSSTR(IGRUP) + 1
      ISSTR(I) =   MAXSTR
      GO TO   60
   50 CONTINUE
      ISSTR(I) =   ISSTR(J)
   60 CONTINUE
      GO TO   90
C
   70 CONTINUE
      INCR       =   MAXSTR
      MAXSTR     =   MAXSTR + NS
      NSSTR(IGRUP) = NSSTR(IGRUP) + NS
      DO   80   I = J1, J2
      INCR       =   INCR + 1
      ISSTR(I) =   INCR
   80 CONTINUE
C
   90 IOLD       =   IGRUP
C
C     WRITE   (6,10)   NS, J2, IOLD, IGRUP, MAXSTR
C     WRITE   (6,20)   (ISSTR(J), J = 1, J2)
C     WRITE   (6,30)   (COORD(J), J = 1, J2)
C
      RETURN
      END
```

```
      SUBROUTINE GUST(  NWT , NWTA  , NTOT, NBOX, NB, NBARAY, NCARAY,
     1           NBEA, FMACH, KR, CBAR, X, YS, ZS, SG, CG, CR, NG,
     2    IPRINT, LINES, NOUT, WGS, WGA, NDW, NMT, NASYM, COL )
      COMPLEX WGR, WGL, WGS(350), WGA(350), COL(350)
      DIMENSION NBARAY(1), NCARAY(1), NBEA(2,1),
     1           X(1), YS(1), ZS(1), SG(1), CG(1), CR(3,1)
      REAL       KR
C
      NTZ        =  (NTOT - NBOX) / 2
      DO  80  NR = 1,NG
      R       = FMACH / (1.0+FMACH*CR(1,NR))
      CONST      =  2.0*KR*R / CBAR
      LS         =  1
      LP         =  1
      NBXS       =  NCARAY(LP)
C
      CALL         ZEROUT(WGS,  NTOT,  1,  0)
      CALL         ZEROUT(WGA,  NTOT,  1,  0)
      II           =  0
      DO  30  I = 1, NBOX
C
      SING       =  SG(LS)
      COSG       =  CG(LS)
C
   10 II = II+1
      THR      = -SING*CR(2,NR) + COSG*CR(3,NR)
      THL      =  SING*CR(2,NR) + COSG*CR(3,NR)
      XL       =  X(II)*CR(1,NR) + ZS(LS)*CR(3,NR)
      YTRM     =  YS(LS)*CR(2,NR)
      XLR        =  XL + YTRM
      XLL        =  XL - YTRM
      ARGR       =  CONST * XLR
      ARGL       =  CONST * XLL
      WGR        =  THR * CMPLX(COS(ARGR), -SIN(ARGR))
      WGL        =  THL * CMPLX(COS(ARGL), -SIN(ARGL))
C
      WGS(I)     =  0.5*(WGR + WGL)
      WGA(I)     =  0.5*(WGR - WGL)
C
      IF (I.LT.NBXS) GO TO 30
      LS         =  LS + 1
C
      IF  (I .EQ. NBARAY(LP))  LP = LP + 1
      NBXS       =  NBXS + NCARAY(LP)
C
   30 CONTINUE
C
      IF  (NB .EQ. 0)  GO TO  39
      NRM1       =  NR - 1
      NPOS       =  NRM1 + 2*NMT + 2
      IF  (NASYM .EQ. 0)  NPOS = NPOS - 1
      REWIND  NDW
```

```
      DO  32  NX = 1, NPOS
   32 READ   (NDW)
C
   33 READ   (NDW)  (COL(I),   I = 1, NTOT)
C
      DO  34  I = 1, NTOT
      WGS(I)   =  WGS(I) - COL(I)
   34 CONTINUE
      IF  (NASYM .EQ. 0)  GO TO  39
      IF (NG.EQ.1) GO TO 37
      NGM1 = NG-1
      DO  36  NX = 1, NGM1
   36 READ   (NDW)
   37 CONTINUE
C
      READ   (NDW)  (COL(I),   I = 1, NTOT)
      DO  38  I = 1, NTOT
      WGA(I)   =  WGA(I) - COL(I)
   38 CONTINUE
C
   39 CONTINUE
C
      WRITE  (NWT  )  (WGS(I),   I = 1, NTOT)
      IF  (NASYM .NE. 0)
     1WRITE  (NWTA )  (WGA(I),   I = 1, NTOT)
      IF  (IPRINT .EQ. 0)  GO TO  70
      IG2      =  0
      LINE3    =  LINES*3
      LPAGE    =  NTOT / LINE3 + 1
      DO  40  KG = 1, LPAGE
      IG1      =  IG2 + 1
      IG2      =  IG2 + LINE3
      IF  (IG2 .GT. NTOT)  IG2 = NTOT
      WRITE  (NOUT,50)  NR, (I, WGS(I),   I = IG1, IG2)
      IF  (NASYM .NE. 0)
     1WRITE  (NOUT,60)  NR, (I, WGA(I),   I = IG1, IG2)
C
   40 CONTINUE
   50 FORMAT   ( 1H1///8H  COLUMN, I5,15H OF SYMM. GUST  //
     1         ( 3 (I6, 2E13.5 ) ) )
   60 FORMAT   ( 1H1///8H  COLUMN, I5,20H OF ANTISYMM. GUST  //
     1         ( 3 (I6, 2E13.5 ) ) )
   70 CONTINUE
C
   80 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE HATS( XA , YA , XO , YO , LMBDA , X14C , X34C , Y    ,  HATS
     1            H14C, H34C, HP   , IHAT , IFPRLL, NPAIR, I1I2 , NTH  ,  HATS
     2            SCOEF,  THB                                          )  HATS
C                                                                          HATS
C        THIS SUBROUTINE COMPUTES ELEMENTS OF THE  H14C-,  H34C-  AND     HATS
C        HP- MATRIX COLUMNS FOR BOXES IN ONE SECTION OF A SUPERPANEL      HATS
C                                                                          HATS
      DIMENSION  X14C(1), X34C(1), Y(1)    , H14C(1), H34C(1), HP(1)   ,  HATS
     1           I1I2(2,1)         , SCOEF(1)         , THB(1)            HATS
      REAL       LMBDA                                                    HATS
C                                                                          HATS
      SINL      =  SIN(LMBDA)                                             HATS
      COSL      =  COS(LMBDA)                                             HATS
      IF  (IFPRLL .EQ. 1)  GO TO  10                                      HATS
      XDIF      =  XA - XO                                                HATS
      YDIF      =     - YO                                                HATS
      RHONOT    =  SQRT(XDIF**2 + YDIF**2)                                HATS
      CALL         ATAN3( YDIF, XDIF, THNOT )                             HATS
C                                                                          HATS
      ROCOS     =  RHONOT * COS(THNOT + LMBDA)                            HATS
CC    WRITE  (6,110)  RHONOT, THNOT, LMBDA, ROCOS,  XA , YA               HATS
   10 CONTINUE                                                            HATS
C                                                                          HATS
      DO  140  IPR = 1, NPAIR                                             HATS
      I1        =  I1I2(1, IPR)                                           HATS
      I2        =  I1I2(2, IPR)                                           HATS
C                                                                          HATS
      DO  130  I = I1, I2                                                 HATS
      IF  (IFPRLL .EQ. 1)  GO TO  20                                      HATS
      X1MXO     =  X14C(I) - XO                                           HATS
      X3MXO     =  X34C(I) - XO                                           HATS
      YMYO      =  Y(I)    - YO                                           HATS
      CALL         ATAN3( YMYO, X1MXO, THI1 )                             HATS
      CALL         ATAN3( YMYO, X3MXO, THI3 )                             HATS
C                                                                          HATS
      ROSIN1    =  RHONOT * SIN(THNOT - THI1)                             HATS
      ROSIN3    =  RHONOT * SIN(THNOT - THI3)                             HATS
      COST1L    =  COS(THI1 + LMBDA)                                      HATS
      COST3L    =  COS(THI3 + LMBDA)                                      HATS
      THA1      = -ROSIN1/ COST1L                                         HATS
      THA3      = -ROSIN3/ COST3L                                         HATS
      GO TO  30                                                           HATS
C                                                                          HATS
   20 CONTINUE                                                            HATS
      YTRM      =   Y(I)         * COSL                                   HATS
      THA1      =  YTRM + (X14C(I) - XA) * SINL                           HATS
      THA3      =  YTRM + (X34C(I) - XA) * SINL                           HATS
   30 CONTINUE                                                            HATS
C                                                                          HATS
      GO TO  (40,50,70),  IHAT                                            HATS
   40 CONTINUE                                                            HATS
      A1        =  1.0                                                    HATS
```

```
      A3        =  1.0
      B         =  0.0
      C         =  0.0
      GO TO  90
   50 CONTINUE
      B         =  -COSL
C
   60 CONTINUE
      IF   (IFPRLL .EQ. 0)   GO TO  72
      YIYASL    =   Y(I)          * SINL
      A1        =  YIYASL - (X14C(I) - XA) * COSL
      A3        =  YIYASL - (X34C(I) - XA) * COSL
      C         =  A3 * SINL
      GO TO  90
C
   70 CONTINUE
      B         =  -SINL
      IF   (IFPRLL .EQ. 1)   GO TO  80
   72 CONTINUE
      RHOB1     =  ROCOS / COSTIL
      RHOB3     =  ROCOS / COST3L
      YIYOSQ    =  (Y(I) - YO)**2
      RHO1      =  SQRT((X14C(I) - XO)**2 + YIYOSQ)
      RHO3      =  SQRT((X34C(I) - XO)**2 + YIYOSQ)
      SINTH1    =  SIN(THI1)
      SINTH3    =  SIN(THI3)
      A1        =  (RHOB1 - RHO1) * COSTIL
      A3        =  (RHOB3 - RHO3) * COST3L
      C         =             (RHO3 - RHOB3) * SINTH3 * RHOB3 / RHO3
      IF   (IHAT .EQ. 2)   GO TO  90
      TANTIL    =  TAN(THI1 + LMBDA)
      TANT3L    =  TAN(THI3 + LMBDA)
      A1        =  A1 * TANTIL
      A3        =  A3 * TANT3L
      C         =  C  * TANT3L
      GO TO  90
   80 A1        =  0.0
      A3        =  0.0
      C         =  0.0
   90 CONTINUE
CC    WRITE  (6,120)   I, X14C(I), X34C(I), Y(I)
CC    WRITE  (6,110)   THA1, THA3, RHOB1, RHOB3, RHO1, RHO3
CC    WRITE  (6,110)   A1 , A3 , B  , C
C
      SUM1      =  0.0
      SUM3      =  0.0
      SUMP      =  0.0
C
      DO  100  IT = 1, NTH
      TDF1      =  THA1 - THB(IT)
      TDF3      =  THA3 - THB(IT)
      TDF1SQ    =  TDF1**2
```

```
      TDF3SQ    =  TDF3**2
      ALTRM1    =  0.0
      IF  (TDF1SQ .GT. .0001) ALTRM1 = ALOG(TDF1SQ)
      ALTRM3    =  0.0
      IF  (TDF3SQ .GT. .0001) ALTRM3 = ALOG(TDF3SQ)
      XMULT1    =  TDF1SQ * ALTRM1
      XMULT3    =  TDF3SQ * ALTRM3
      XMULTP    =  TDF3 * ( ALTRM3 + 1.0 )
C
      IP2       =  IT + 2
      SUM1      =  SUM1 + XMULT1 * SCOEF(IP2)
      SUM3      =  SUM3 + XMULT3 * SCOEF(IP2)
      SUMP      =  SUMP + XMULTP * SCOEF(IP2)
C
  100 CONTINUE
      CMBR1     =  SCOEF(1) + SCOEF(2) * THA1 + SUM1
      CMBR3     =  SCOEF(1) + SCOEF(2) * THA3 + SUM3
      SLOPE     =  SCOEF(2) + 2.0 * SUMP
CC    WRITE  (6,110)  CMBR1, CMBR3, SLOPE
  110 FORMAT  ( /   6E15.6 )
  120 FORMAT  ( / I15, 5E15.6 )
C
      H14C(I)   =  CMBR1 * A1
      H34C(I)   =  CMBR3 * A3
      HP(I)     =  CMBR3 * B + SLOPE * C
  130 CONTINUE
C
  140 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE IDF1 (EE,E2,     ETA01,ZET01,ARE,AIM,BRE,BIM,CRE,CIM,
     1                 R1SQX,XIIJR,XIIJI)
C    ***   INTEGRATES THE PLANAR PARTS OF THE INCREMENTAL
C          OSCILLATORY KERNELS FOR UNSTEADY CASES
   10 FORMAT  (1H0,    6E20.6/)
      PI   = 3.1415926
      PARN = ETA01**2 - ZET01**2
      FACR = PARN*ARE + ETA01*BRE + CRE
      FACI = PARN*AIM + ETA01*BIM + CIM
      PARNR= BRE/2.0   + ETA01*ARE
      PARNI= BIM/2.0   + ETA01*AIM
      UP   = (ETA01-EE)**2 + ZET01**2
      DOWN = (ETA01+EE)**2 + ZET01**2
      ARG2 = UP/DOWN
      ALARG2 = ALOG(ARG2)
      TRM2R= PARNR * ALARG2
      TRM2I= PARNI * ALARG2
      TRM3R= 2.0*EE* ARE
      TRM3I= 2.0*EE* AIM
      AZET = ABS(ZET01)
      IF  ((AZET/EE) . LE . 0.001)  GO TO  100
      TESTO= ABS((R1SQX-E2)/(2.0*EE*AZET))
      IF (TESTO.LE.0.0001)  GO TO 110
      COEF = (2.0*EE)/(R1SQX-E2)
      ARGA = COEF*ZET01
      TEST = ABS(ARGA)
      IF (TEST.LE.0.3)  GO TO 120
      ARGT = COEF*AZET
      ATANA= ATAN(ARGT)
      FUNCT= ATANA/AZET
      GO TO 170
  100 CONTINUE
      FUNCT= (2.0*EE)/(ETA01**2-E2)
      GO TO 170
  110 CONTINUE
      FUNCT= 0.0
      GO TO 170
  120 CONTINUE
      S    = ARGA**2
      SER  = 1./3.+S*(-1./5.+S*(1./7.+S*(-1./9.+S*(1./11.-S/13.)))))
      ALPHA= E2*(COEF**2)*SER
      FUNCT= COEF*(1.0-ALPHA*(ZET01**2)/E2)

  170 CONTINUE
      TRM1R= FACR * FUNCT
      TRM1I= FACI * FUNCT
      XIIJR= TRM1R + TRM2R + TRM3R
      XIIJI= TRM1I + TRM2I + TRM3I
      RETURN
      END
```

```
      SUBROUTINE IDF2(EE,E2,     ETA01,ZET01,A2R,A2I,B2R,B2I,C2R,C2I,
     1               R1SQX,DIIJR,DIIJI)
C    ***   INTEGRATES THE NONPLANAR PARTS OF THE INCREMENTAL
C          OSCILLATORY KERNELS FOR UNSTEADY CASES
   10 FORMAT  (1H0,    6E20.6/)
      EPS   = 0.0001
      AZET  = ABS(ZET01)
      DENO  = R1SQX-E2
      PARN  = ETA01**2  + ZET01**2
      FACR  = PARN*A2R  + ETA01*B2R  + C2R
      FACI  = PARN*A2I  + ETA01*B2I  + C2I
      ETA02=ETA01**2
      ZET02= ZET01**2
      IF   ((AZET/EE) . LE . 0.001)  GO TO  120
      TESTO= ABS((R1SQX-E2)/(2.0*EE*AZET))
      IF  (TESTO.GT.0.1)  GO TO 120
      DEN2 = (ETA01+EE)**2+ZET02
      DEN3 = (ETA01-EE)**2+ZET02
      FAC2A= R1SQX*ETA01+(ETA02-ZET02)*EE
      FAC3A= R1SQX*ETA01-(ETA02-ZET02)*EE
      FAC2B= R1SQX+ETA01*EE
      FAC3B= R1SQX-ETA01*EE
      TRM2R= (FAC2A*A2R+FAC2B*B2R+(ETA01+EE)*C2R)/DEN2
      TRM2I= (FAC2A*A2I+FAC2B*B2I+(ETA01+EE)*C2I)/DEN2
      TRM3R=-(FAC3A*A2R+FAC3B*B2R+(ETA01-EE)*C2R)/DEN3
      TRM3I=-(FAC3A*A2I+FAC3B*B2I+(ETA01-EE)*C2I)/DEN3
      IF  (TESTO.LE.0.0001)  GO TO 110
      COEF = (2.0*EE)/(R1SQX-E2)
      ARGA = COEF*ZET01
      TEST = ABS(ARGA)
      IF   (TEST.GT.0.3)  GO TO 90
      S     = ARGA**2
      SER   = 1./3.+S*(-1./5.+S*(1./7.+S*(-1./9.+S*(1./11.-S/13.))))
      ALPHA= E2*(COEF**2)*SER
      FUNCT= COEF*(1.0-ALPHA*(ZET01**2)/E2)
      GO TO 100
   90 CONTINUE
      ARGT = COEF*AZET
      ATANA= ATAN(ARGT)
      FUNCT= ATANA/AZET
  100 CONTINUE
      TRM1R= FACR*FUNCT
      TRM1I= FACI*FUNCT
      DIIJR= (TRM1R  + TRM2R  +  TRM3R)/(2.0*ZET02)
      DIIJI= (TRM1I  + TRM2I  +  TRM3I)/(2.0*ZET02)
      GO TO 170
  110 CONTINUE
      FUNCT= 0.0
      GO TO 100
  120 CONTINUE
      DENA = (ETA01+EE)**2  + ZET01**2
      DENB = (ETA01-EE)**2  + ZET01**2
```

```
      UP1R = 2.0*(E2*A2R + C2R)                                               IDF2
      UP1I = 2.0*(E2*A2I + C2I)                                               IDF2
      UP2R = 4.0*E2*ETA01*B2R                                                 IDF2
      UP2I = 4.0*E2*ETA01*B2I                                                 IDF2
      TRM1R= (UP1R *(R1SQX+E2) + UP2R )/(DENA*DENB)                           IDF2
      TRM1I= (UP1I *(R1SQX+E2) + UP2I )/(DENA*DENB)                           IDF2
      IF  ((AZET/EE) . LE . 0.001)  GO TC  130                                IDF2
      COEF = (2.0*EE)/(R1SQX-E2)                                              IDF2
      ARGA = COEF*ZET01                                                       IDF2
      TEST = ABS(ARGA)                                                        IDF2
      IF  (TEST.GT.0.3)  GO TO 125                                            IDF2
      S    = ARGA**2                                                          IDF2
      SER  = 1./3.+S*(-1./5.+S*(1./7.+S*(-1./9.+S*(1./11.-S/13.))))           IDF2
      ALPHA= E2*(COEF**2)*SER                                                 IDF2
      FUNCT= COEF*(1.0-ALPHA*(ZET01**2)/E2)                                   IDF2
      GO TO 140                                                               IDF2
  125 CONTINUE                                                                IDF2
      ARGT= COEF*AZET                                                         IDF2
      ATANA= ATAN(ARGT)                                                       IDF2
      FUNCT= ATANA/AZET                                                       IDF2
      ALPHA= (E2/ZET02)*(1.0-FUNCT*(DENO/(2.0*EE)))                           IDF2
      GO TO 140                                                               IDF2
  130 CONTINUE                                                                IDF2
      ALPHA= ((2.0*E2)/(ETA02-E2))**2                                         IDF2
  140 CONTINUE                                                                IDF2
      TRM2R= -ALPHA*FACR/E2                                                   IDF2
      TRM2I= -ALPHA*FACI/E2                                                   IDF2
      DIIJR= EE*(TRM1R + TRM2R)/DENO                                          IDF2
      DIIJI= EE*(TRM1I + TRM2I)/DENO                                          IDF2
  170 CONTINUE                                                                IDF2
      RETURN                                                                  IDF2
      END                                                                     IDF2
```

```
      SUBROUTINE INCRO(AX,AY,AZ,AX1,AY1,AZ1,AX2,AY2,AZ2,SGR,CGR,SGS,CGS,
     1          KR,FL,BETA,SDELX,DFLY,DELR,DELI,IO,IR,NBXS,NCPNB,LHS,
     2          NDBLE,IMG,NOBI,IMGS,  USE1 ,      USE2,       USE3,
     3          USE4,      XUSE1,     XUSE2,      XUSE3,      XUSE4 )
C    ***   CALCULATES THE UNSTEADY PART OF THE INFLUENCE COEFFICIENT
C          MATRIX ELEMENTS USING  SUBROUTINES  KERNEL, IDF1  AND  IDF2
      REAL    K10,K20,K1RT1,K1IT1,K2RT2P,K2IT2P,K10T1,K20T2P, KR,M
      COMMON  /DLM/  K10,K20,K1RT1,K1IT1,K2RT2P,K2IT2P,K10T1,K20T2P,E2
      COMMON       /KDS/ IND,KD1R,KD1I,KD2R,KD2I
      REAL         KD1R,KD1I, KD2R,KD2I
      DIMENSION    USE1(14,1),  USE2(14,1),  USE3(14,1),  USE4(14,1)
      DIMENSION   XUSE1(14,1), XUSE2(14,1), XUSE3(14,1), XUSE4(14,1)
    6 FORMAT (1H0,15I5)
   10 FORMAT (1H0,6F20.8)
C*********************************************************************
C
C    *****    THE FOLLOWING LIST IS THE NAMES OF THE KERNEL VALUES
C    *****    USED IN THE SAVING PROCESS OF KERNELS
C
C    **   DKRO = REAL PART OF THE PLANAR KERNEL   *  OUTBOARD POINT
C    **   DKIO =  IMAGINARY PART OF THE PLANAR KERNEL   *   OUTBOARD POINT
C    **   XKRO = REAL PART OF THE NONPLANAR KERNEL   *  OUTBOARD POINT
C    **   XKIO =  IMAGINARY PART OF THE NONPLANAR KERNEL   *   OUTBOARD POINT
C    **   DKRI = REAL PART OF THE PLANAR KERNEL   *    INBOARD POINT
C    **   DKII =  IMAGINARY PART OF THE PLANAR KERNEL   *    INBOARD POINT
C    **   XKRI = REAL PART OF THE NONPLANAR KERNEL   *    INBOARD POINT
C    **   XKII =  IMAGINARY PART OF THE NONPLANAR KERNEL   *    INBOARD POINT
C
C    *****    THE FOLLOWING TABLE EXPLAINS THE USAGE OF THE
C    *****    STORED KERNEL ARRAYS
C
C*********************************************************************
C
C   SENDING POINT IN     *  SENDING POINT IN      *  VALUE OF INDEX 'J' WHEN
C   RIGHT HALF-PLANE     *  LEFT  HALF-PLANE      *       SENDING POINT IS
C     (Y.GE.0)           *     (Y.LT.0)           *......................
C........................*.......................* ON LIFTING . INSIDE ASSOC.
C                        *                        *   SURFACE   .  BODY (NOBI)
C'OUTBOARD'.   SAVED     * 'INBOARD'.   SAVED     *......................
C  KERNELS .    IN       *  KERNELS .    IN       *UPPER* LOWER. UPPER * LOWER
C          .             *          .             * HALF-PLANE . HALF-PLANE
C          .             *          .             *     .      .       .
C*********************************************************************
C          .             *          .             *
C   DKRO   . USE1(I,J)*   DKRI     . USE3(I,J)*        .      .       .
C   DKIO   . USF2(I,J)*   DKII     . USE4(I,J)*        .      .       .
C          .             *          .             *  1   .  2   .2+NOBI  .8+NCBI
C   XKRO   .XUSE1(I,J)*   XKRI     .XUSE3(I,J)*        .      .       .
C   XKIO   .XUSF2(I,J)*   XKII     .XUSF4(I,J)*        .      .       .
C          .             *          .             *     .      .       .
C*********************************************************************
C
```

```
      IND   = 1
      M = SQRT(1.0 - BETA**2)
      BR   = FL/2.
      EPS = 0.00001
      PI    = 3.14159265
      XDELX =  SDELX
      XDELY =   DELY
      EE = 0.5*XDELY
      E2    = EE**2
      DELR = 0.0
      DELI = 0.0
      T1S  = 0.0
      T2S  = 0.0
      AT1S = 0.0
      AT2S = 0.0
      T1    = 0.0
      T2  = 0.0
      COUNT =0.
      XO=AX
      YO=AY
      ZO=AZ
   80 CONTINUE
      CALL        TKER   (XO,YO,ZO,KR,BR,SGR,CGR,SGS,CGS,T1,T2,M)
      AT1   = ABS(T1)
      AT2   = ABS(T2)
      IF  (AT1.GT.AT1S)   AT1S=AT1
      IF  (AT2.GT.AT2S)   AT2S=AT2
      IF  (COUNT)130,90,150
   90 DKRC  = K1RT1 - K1OT1
      DKIC  = K1IT1
      XKRC  = K2RT2P-K2OT2P
      XKIC  = K2IT2P
      AT2 = ABS(T2)
      JO = 1
      IF   (IMG.NE.0)  JO = 2 + NOBI
      IF  (NDBLE.NE.0) JO = 2
      IF   (IMG.NE.0.AND.NDBLE.NE.0)   JO = 8 + NOBI
      IF  (IR.LE.NCPNB) GO TO  110
      IF  (IR.GT.NBXS) GO TO 110
      IF  (LHS.NE.0) GO TO  100
      DKRI = USE1(JO,IO)
      DKII = USE2(JO,IO)
      XKRI  = XUSE1(JO,IO)
      XKII  = XUSE2(JO,IO)
      XO    = AX2
      YO    = AY2
      ZO  = AZ2
      CALL        TKER   (XO,YO,ZO,KR,BR,SGR,CGR,SGS,CGS,T1,T2,M)
      AT1   = ABS(T1)
      AT2   = ABS(T2)
      IF  (AT1.GT.AT1S)   AT1S=AT1
      IF  (AT2.GT.AT2S)   AT2S=AT2
```

```
      DKRO   = K1RT1 - K1OT1
      DKIO   = K1IT1
      XKRO   = K2RT2P-K2OT2P
      XKIO   = K2IT2P
      GO TO  170
  100 XO = AX1
      YO = AY1
      ZO = AZ1
      CALL        TKER  (XO,YO,ZO,KR,BR,SGR,CGR,SGS,CGS,T1,T2,M)
      AT1   = ABS(T1)
      AT2   = ABS(T2)
      IF (AT1.GT.AT1S)  AT1S=AT1
      IF (AT2.GT.AT2S)  AT2S=AT2
      DKRI   = K1RT1 - K1OT1
      DKII   = K1IT1
      XKRI   = K2RT2P-K2OT2P
      XKII   = K2IT2P
      DKRO = USE3(JO,IO)
      DKIO = USE4(JO,IO)
      XKRO   = XUSE3(JO,IO)
      XKIO   = XUSE4(JO,IO)
      GO TO  160
  110 CONTINUE
      COUNT =-1.
  120 XO = AX1
      YO = AY1
      ZO = AZ1
      GO TO 80
  130 DKRI   = K1RT1 - K1OT1
      DKII   = K1IT1
      XKRI   = K2RT2P-K2OT2P
      XKII   = K2IT2P
  140 COUNT  = 1.
      XO=AX2
      YO=AY2
      ZO=AZ2
      GO TO 80
  150 DKRO   = K1RT1 - K1OT1
      DKIO   = K1IT1
      XKRO   = K2RT2P-K2OT2P
      XKIO   = K2IT2P
      JO = 1
      IF  (IMG.NE.0)  JO = 2 + NOBI
      IF (NDBLE.NE.0) JO = 2
      IF  (IMG.NE.0.AND.NDBLE.NE.0)  JO = 8 + NOBI
      IF (LHS.EQ.0) GO TO  170
  160 USE3(JO,IO) = DKRI
      USE4(JO,IO) = DKII
      XUSE3(JO,IO)= XKRI
      XUSE4(JO,IO)= XKII
      GO TO 180
  170 USE1(JO,IO) = DKRO
```

```
      USE2(JO,IO) = DKIO                                                 INCRO158
      XUSE1(JO,IO)= XKRO                                                 INCRO159
      XUSE2(JO,IO)= XKIO                                                 INCRO160
  180 CONTINUE                                                           INCRO161
      XO = AX                                                            INCRO162
      YO = AY                                                            INCRO163
      ZO = AZ                                                            INCRO164
      ZERO = 0.0                                                         INCRO165
      XIIJR= 0.                                                          INCRO166
      XIIJI= 0.                                                          INCRO167
      DIIJR= 0.0                                                         INCRO168
      DIIJI= 0.0                                                         INCRO169
      PI    = 3.1415926                                                  INCRO170
      XMULT =((XDELX)/(8.0*PI))                                          INCRO171
      IF ((YO.EQ.ZERO).AND.(ZO.EQ.ZERO))  GO TO 220                      INCRO172
      IF    ((ZO.EQ.ZERO).AND.(SGS.EQ.ZERO))   GO TO 230                 INCRO173
  190 ETAO1 = YO*CGS + ZO*SGS                                            INCRO174
      ZETO1 =-YO*SGS + ZO*CGS                                            INCRO175
      AZETO = ABS(ZETO1)                                                 INCRO176
      IF (AZETO.LE.0.0001)  ZETO1 = 0.                                   INCRO177
  200 RISQX = ETAO1**2 + ZETO1**2                                        INCRO178
  210 ARE = (DKRI -2.*DKRC + DKRO)/(2.0*E2)                              INCRO179
      AIM = (DKII -2.*DKIC + DKIO)/(2.0*E2)                              INCRO180
      BRE = (DKRO - DKRI)/(2.0*EE)                                       INCRO181
      BIM = (DKIO - DKII)/(2.0*EE)                                       INCRO182
      CRE =  DKRC                                                        INCRO183
      CIM =  DKIC                                                        INCRO184
      GO TO 250                                                          INCRO185
  220 ETAO1 = 0.0                                                        INCRO186
      ZETO1 = 0.0                                                        INCRO187
      RISQX = 0.0                                                        INCRO188
      GO TO  210                                                         INCRO189
  230 ETAO1  = YO*CGS                                                    INCRO190
      ZETO1  = 0.                                                        INCRO191
  240 RISQX  = ETAO1**2                                                  INCRO192
      GO TO  210                                                         INCRO193
  250 CONTINUE                                                           INCRO194
      IF (AT1S.EQ.0.0)  GO TO 255                                        INCRO195
      CALL        IDF1 (EE,E2,     ETAO1,ZETO1,ARE,AIM,BRE,BIM,CRE,CIM,  INCRO196
     1                 RISQX,XIIJR,XIIJI)                                INCRO197
      DELR =XMULT*XIIJR                                                  INCRO198
      DELI =XMULT*XIIJI                                                  INCRO199
  255 CONTINUE                                                           INCRO200
      IF (AT2S.EQ.0.0)  GO TO 260                                        INCRO201
      A2R  = (XKRI - 2.0*XKRC + XKRO)/(2.0*E2)                           INCRO202
      A2I  = (XKII - 2.0*XKIC + XKIO)/(2.0*E2)                           INCRO203
      B2R  = (XKRO - XKRI)/(2.0*EE)                                      INCRO204
      B2I  = (XKIO - XKII)/(2.0*EE)                                      INCRO205
      C2R  =  XKRC                                                       INCRO206
      C2I  =  XKIC                                                       INCRO207
      CALL        IDF2(EE,E2,     ETAO1,ZETO1,A2R,A2I,B2R,B2I,C2R,C2I,   INCRO208
     1                RISQX,DIIJR,DIIJI)                                 INCRO209
```

```
      DELR = DELR  + XMULT*DIIJR                                       INCR0210
      DELI = DELI  + XMULT*DIIJI                                       INCR0211
  260 CONTINUE                                                         INCR0212
C    ***                                                               INCR0213
C    ***                                                               INCR0214
  265 RETURN                                                           INCR0215
      END                                                              INCR0216
```

```
      SUBROUTINE INPUTA (NTI, NMS, NDOF, MODES, IPR
     1 , ELXI, ELYI, ELZI, PHI )
C
      DIMENSION ELXI(1), ELYI(1), ELZI(1)
      DIMENSION PHI(NDOF,NMS,MODES)
C
      COMMON /ZZZ/HEDR(48),NIN,NOUT,KROW,LINES
      DIMENSION DEFL(8)
      DATADEFL/3H  F,3H  L,3H  H,5HTHETA,5HALPHA,4H PSI,5H BETA,5HDELTA/
      DATA AERO/4HAERO/
C
   10 FORMAT (6F12.0)
   20 FORMAT (1H0,20X,*MODESHAPE FOR MODE*,I4,4X5HFREQ=,E11.4
     1 / 5HOSECT,6X,A5,8(9XA5) )
   21 FORMAT (I5,9E14.6)
C
      REWIND NTI
      READ (NTI)
      READ (NTI)
      CALL HEADNG
      WRITE (NOUT,50)
   50 FORMAT (1H0,4HNODE,8X7HELXI(I),8X7HELYI(I),8X7HELZI(I)/)
   40 FORMAT (1X,I4,3E15.6)
      DO 100 I=1,NMS
      READ (NTI,10) ELXI(I), ELYI(I), ELZI(I)
  100 WRITE (NOUT,40) I, ELXI(I), ELYI(I), ELZI(I)
C
C     SKIP BY FREQUENCIES
      READ (NTI,10) (DUM,I=1,MODES)
C
      DO 130 K=1,MODES
      LINES = KROW
      DO 120 J=1,NMS
      READ (NTI, 10) (PHI(I,J,K),I=1,NDOF)
      IF (IPR.EQ.0) GO TO 120
      IF (LINES.LT.KROW) GO TO 115
      CALL HEADNG
      WRITE (NOUT, 20) K, WR    , (DEFL(I),I=1,NDOF)
      LINES = LINES+4
  115 WRITE (NOUT, 21) J, (PHI(I,J,K),I=1,NDOF)
      LINES = LINES+1
  120 CONTINUE
  130 CONTINUE
C
  200 READ (NTI,90) CHECK
   90 FORMAT (A4)
      IF (CHECK.NE.AERO) GO TO 200
C
      READ (NTI)
      READ (NTI)
      READ (NTI)
C
```

```
      RETURN                                                            INPUTA54
      END                                                               INPUTA55
```

```
      SUBROUTINE MATMUL(NW, NPSTAP, NBFM, NEWBFZ, NEWBFY, NWORK, NTSBE,
     1               NSBETO, LENGTH, NTOT, NMODE, NMODEB, IPRINT,  KR ,
     2               DT     , RDT   ,   WORK,   RWORK                   )
C
      COMPLEX     WORK(NWORK),  DT(900)
      DIMENSION   RWORK(2,NWORK), RDT(2,900)
      REAL        KR
C
   10 FORMAT  (1HO///25X, I4,44H  SLENDER BODY ELEMENT FORCES   **  MODF
     1 NO., I4// 20X, 11HELEMENT NO., 12X, 2HFZ, 30X, 2HFY/)
   20 FORMAT   (      25X, I4, 4E16.8)
   25 FORMAT   ( 1H1 /// )
      REWIND  NW
      REWIND  NBFM
      REWIND  NEWBFZ
      REWIND  NEWBFY
      NSBET2 = 2 * NSBETO
      NBEG      =  0
      NBEG1   = NBEG + 1
      NBEG2   = NTOT + 1
      IFLAG     =  1
      IBEG = 2*LENGTH+1
      DO  30   IM=1,NWORK
   30 WORK(IM) = 0.0
      DO  40   IM = 1, NSBET2
      CALL          RWREC(IFLAG, NBFM, WORK(IBEG), LENGTH, 1, 0 )
      IBEG = IBEG + LENGTH
C
   40 CONTINUE
      L         =  0
      DO  90  J=1,NMODE
      IF  (IPRINT .GE. 2)  WRITE (6,10)  NSBETO, J
      IFLAG   = 1
      CALL          ZEROUT( DT, LENGTH,  1 ,  0 )
C
C  ***   READ ONE COLUMN OF SOLUTIONS  FROM TAPE  NW
C
      CALL          RWRFC(IFLAG,   NW  , DT(  1  ),  NTOT  , 1, 0  )
C
C  ***   APPEND THE ARRAY CONTAINING  ONE COLUMN OF SOLUTIONS  BY THE
C        CORRESPONDING COLUMN OF CP-Z-DELTA-A  AND CP-Y-DELTA-A  **
C        THESE ARE READ FROM TAPE  NPSTAP
C
      CALL          RWREC(IFLAG, NPSTAP, DT(NBEG2),   NTSBE, 1, 0  )
      KK = 2*LENGTH
      DO  50  IM = 1, NSBET2
   50 WORK(IM) = 0.0
      DO  70  I =1,NSBET2
C
      NBW  = NBEG + I
      DO  60  K=1, LENGTH
      KK        =  KK + 1
```

```
      RWORK(1,NBW) = RWORK(1,NBW) + RWORK(1,KK) * RDT(1,K)
     1                            - RWORK(2,KK) * RDT(2,K)
      IF  (KR . EQ . 0.0)  GO TO  60
C
      RWORK(2,NBW) = RWORK(2,NBW) + RWORK(2,KK) * RDT(1,K)
     1                            + RWORK(1,KK) * RDT(2,K)
   60 CONTINUE
   70 CONTINUE
      IFLAG  = 0
      WRITE  (NEWBFZ)  (WORK(IWK),  IWK = 1, NSBET2, 2)
      WRITE  (NEWBFY)  (WORK(IWK),  IWK = 2, NSBFT2, 2)
C
      IF  (IPRINT .LT. 2)  GO TO  90
      IWY    = NBEG1 - 1
      DO  80  K = 1, NSBETO
      L        =  L + 1
      IWZ    = IWY + 1
      IWY    = IWZ + 1
      WRITE  (6,20)  K, WORK(IWZ), WORK(IWY)
   80 CONTINUE
   90 CONTINUF
      RETURN
      END
```

```
      SUBROUTINE  MUZYC ( NMODE,          K, NTZY, NFBODY, NLBODY, NSBE,
     1              KR      ,          CBAR   , AO     , AOP    , XIS1   ,XIS2,
     2    AR, NTPH, NTPDH, UZY, CPZY,
     3         UZYA, CPZYA, NR, FMACH, YB, ZB, CR, NG, H, DHDX )
CMUZYC          CALCULATION OF MU-Z AND MU-Y  COLUMNS
C               USED TO DETERMINE NORMALWASH FLOW FIELD DUE TO SLENDER
C               BODIES
C
C
C  ***       *     *     *     *     *     *     *     *     *     *     *     *     *
C
C            NMODE        MODE NUMBER REQUESTED
C            K            FLAG FOR TYPE OF COEFF.     = 2 - Z,  = 3 - Y
C            NTZY         NUMBER OF ELEMENTS - EITHER = NTZS OR NTYS
C            NFBODY       NUMBER OF FIRST BODY
C            NLBODY       NUMBER OF LAST BODY
C            NSBE         ARRAY OF NUMBER OF SLENDER BODY ELEMENTS PER BODY
C            KR           REDUCED FREQUENCY
C            XS           SLENDER BODY MIDPOINT
C            BR           SEMI CHORD
C            WS           OUTPUT COMPLEX COLUMN
      DIMENSION     UZY( 2, 1 ),    CPZY(2, 1 ), NSBF(10, 2), YB(1),ZB(1),
     1              H(350) , DHDX(350) , CR(3,1) ,
     2               AOP(1) ,   XIS1(1),   XIS2(1),   AR(1)   ,    AO(1)
      COMPLEX     UZYA( 1 ), CPZYA( 1 ), WGR, WGL, WGS, WGA
      REAL          KR
      DATA       TWOPI / 6.283185 /
      DATA           PI / 3.141593 /
C
      TWOKR           =      2.0   * KR
      TPKRC           =   TWOPI * KR / CBAR
      RAR             =   1.0
      LENGTH    =  NTZY + NLBODY-NFBODY + 1
      IF   (NR .NE. 0)   GO TO   20
      DO   10   I = 1, LENGTH
      H    (I)     = 0.0
   10 DHDX(I)      = 0.0
      READ    (NTPH )   ( H   (L),   L = 1, LENGTH)
      READ    (NTPDH)   (DHDX(L),    L = 1, LENGTH)
   20 DO      30    I = 1,   NTZY
      UZY ( 1, I)    =    0.0
      UZY ( 2, I)    =    0.0
      CPZY(1, I)     =        0.0
      CPZY(2, I)     =        0.0
      IF   (NR .EQ. 0)   GO TO    30
      UZYA( I) =   (0.0, 0.0)
      CPZYA(I) =   (0.0, 0.0)
   30 CONTINUE
C
C                       CALCULATE VALUES FOR THE
C                       FIRST SLENDER BODY ELEMENT
      IH1         = 0
```

```
      NFSBEB   =  1                                                      MUZYC  54
      NLSBEB   =  0                                                      MUZYC  55
      NFU         =     0                                                MUZYC  56
      NL       =  NFBODY - 1                                             MUZYC  57
      IF ( NL .LE. 0 )   GO TO  50                                       MUZYC  58
C                   THE FIRST BODY IS NOT = 1, WE MUST, THEREFORE        MUZYC  59
C                   DETERMINE THE LOCATION OF THE FIRST ELEMENT IN THE   MUZYC  60
C                   FIRST BODY.                                          MUZYC  61
      DO     40    I = 1, NL                                             MUZYC  62
      NLSBEB   =      NLSBEB   +  NSBE(   I,   1 )                       MUZYC  63
      IH1      =   IH1 + NLSBEB + 1                                      MUZYC  64
C                                                                        MUZYC  65
   40 NFSBEB          =      NFSBEB   +  NSBE ( I, 1 )                   MUZYC  66
C                                                                        MUZYC  67
C                   LOOP FOR EACH BODY                                   MUZYC  68
   50 CONTINUE                                                           MUZYC  69
      IF   (NR  .EQ. 0)   GO TO   52                                     MUZYC  70
      R         = FMACH / (1. + FMACH * CR(1,NR) )                       MUZYC  71
      CONST     =  TWOKR * R / CBAR                                      MUZYC  72
      CRMULT = 0.5 * CR(3,NR)                                            MUZYC  73
      IF   (K  .FQ.  3)  CRMULT  = 0.5 * CR(2,NR)                        MUZYC  74
   52 CONTINUE                                                           MUZYC  75
      DO      100    IB = NFBODY, NLBODY                                 MUZYC  76
   60 NLSBEB          =      NLSBEB   +  NSBE (IB, 1 )                   MUZYC  77
C                                                                        MUZYC  78
C                 CALCULATE TERMS FOR EACH ELEMENT IN BODY               MUZYC  79
      OPAR      =     1.0  +  AR(IB)                                     MUZYC  80
      DEN       =     TWOPI * OPAR                                       MUZYC  81
      IF  ( K .NE. 3 )  GO TO   70                                       MUZYC  82
      DEN       =     DEN * AR(  IB  )                                   MUZYC  83
      RAR           =   AR(IB)                                           MUZYC  84
   70 CONTINUE                                                           MUZYC  85
      IU        =     NFU                                                MUZYC  86
      DO     80     IT =  NFSBEB, NLSBEB                                 MUZYC  87
      IU        =      IU  +  1                                          MUZYC  88
      WSR           =  0.0                                               MUZYC  89
      WSI           =  0.0                                               MUZYC  90
      WSPR          =  0.0                                               MUZYC  91
      WSPI          =  0.0                                               MUZYC  92
      TWOAO     =     2.0 * AO(IT) * OPAR * ( XIS2(IT) - XIS1(IT)   )    MUZYC  93
      D2D       =     1.0 / ( DEN * AO(IT) **2)                          MUZYC  94
      ELR           =  TWOPI * AOP(IT) * RAR                             MUZYC  95
      ELI           =  TPKRC * AO(IT)  * RAR                             MUZYC  96
      EM        =     PI    *     AO(IT)    *     RAR   /    CBAR        MUZYC  97
      IH1       = IH1 + 1                                                MUZYC  98
      IH2       = IH1 + 1                                                MUZYC  99
      DELXB     = XIS2(IT) - XIS1(IT)                                    MUZYC100
      IF   (NR .NE. 0)   GO TO   72                                      MUZYC101
      WSR       = 0.5 * (DHDX(IH1) + DHDX(IH2))                          MUZYC102
      WSI       = ( H (IH1) + H (IH2) ) * KR / CBAR                      MUZYC103
      WSPR      = (DHDX(IH2) - DHDX(IH1)) * CBAR / DELXB                 MUZYC104
      WSPI      = 2.0 * KR * WSR                                         MUZYC105
```

```
C                                                                        MUZYC106
C         NOTE THAT  WS-PRIME = (WSPR, WSPI)  IS PARTIAL WITH RESPECT    MUZYC107
C         TO   X / CBAR                                                  MUZYC108
C                                                                        MUZYC109
      UZY ( 1,  IU) =  WSR  /  D2D                                       MUZYC110
      UZY ( 2, IU) =  WSI  /  D2D                                        MUZYC111
      CPZY( 1, IU) =  WSR * ELR  -  WSI * ELI  +  WSPR * EM              MUZYC112
      CPZY( 2, IU) =  WSR * ELI  +  WSI * ELR  +  WSPI * EM              MUZYC113
C       CALCULATE  DELTA-CP  * AREA                                      MUZYC114
      CPZY( 1, IU) = -CPZY( 1, IU) * TWOAO                               MUZYC115
      CPZY( 2, IU) = -CPZY( 2, IU) * TWOAO                               MUZYC116
      GO TO  80                                                          MUZYC117
   72 CONTINUE                                                           MUZYC118
      X          =  0.5 * (XIS1(IT) + XIS2(IT) )                         MUZYC119
      XL       = X * CR(1,NR) + ZB(IB) * CR(3,NR)                        MUZYC120
      YTRM     = YB(IB) * CR(2,NR)                                       MUZYC121
      XLR        =  XL + YTRM                                            MUZYC122
      XLL        =  XL - YTRM                                            MUZYC123
      ARGR       =  CONST * XLR                                          MUZYC124
      ARGL       =  CONST * XLL                                          MUZYC125
      WGR        =  CMPLX( COS(ARGR), -SIN(ARGR) )                       MUZYC126
      WGL        =  CMPLX( COS(ARGL), -SIN(ARGL) )                       MUZYC127
      IF  (K .EQ. 3)  GO TO  74                                          MUZYC128
      WGS        =  CRMULT * (WGR + WGL)                                 MUZYC129
      WGA        =  CRMULT * (WGR - WGL)                                 MUZYC130
      GO TO  76                                                          MUZYC131
   74 CONTINUE                                                           MUZYC132
      WGS        =  CRMULT * (WGR - WGL)                                 MUZYC133
      WGA        =  CRMULT * (WGR + WGL)                                 MUZYC134
   76 CONTINUE                                                           MUZYC135
C                                                                        MUZYC136
      GRE        =  REAL (WGS)                                           MUZYC137
      GIM        =  AIMAG(WGS)                                           MUZYC138
      UZY(1,IU)=  GRE / D2D                                              MUZYC139
      UZY(2,IU)=  GIM / D2D                                              MUZYC140
      UZYA( IU)=  WGA / D2D                                              MUZYC141
C                                                                        MUZYC142
      PRE        =  ELR                                                  MUZYC143
      PIM      = ELI - TWOKR * R*CR(1,NR) * EM                           MUZYC144
      CPZY(1,IU)  =  TWOAO * (GRE*PRE - GIM*PIM)                         MUZYC145
      CPZY(2,IU)  =  TWOAO * (GRE*PIM + GIM*PRE)                         MUZYC146
      CPZYA( IU)  =  TWOAO * WGA * CMPLX(PRE, PIM)                       MUZYC147
C                                                                        MUZYC148
   80 CONTINUE                                                           MUZYC149
C                    INCREMENT ELEMENT NUMBER                            MUZYC150
      IH1        =  IH2                                                  MUZYC151
   90 NFU        =  NFU  +  NSBE ( IB, 1 )                               MUZYC152
  100 NFSBEB        =     NFSBEB  +  NSBE (IB, 1 )                       MUZYC153
C                         CALCULATIONS COMPLETED                         MUZYC154
  110 CONTINUE                                                           MUZYC155
      RETURN                                                             MUZYC156
      END                                                                MUZYC157
```

```
      SUBROUTINE NACL(NRUN , NTPSH, NSBETO, NBCUM, X3L    , NXQ , XIQ  ,
     1               NSUP,INSUP,   H   , DHDX   , NMODE, KPRINT,      NTP6 )
C
C          COMPUTES  *H-,  ALPHA-DEGREES-OF-FREEDOM* AT EACH  (AERO-)
C          NACELLE NODE
C
      DIMENSION  NBCUM(1), X3L(1), XIQ(1), INSUP(1), H(1), DHDX(1)
C
   40 FORMAT  ( 6E15.6 )
   50 FORMAT  ( //    6H   THE, I4, 37H  H-ARRAY ELEMENTS FOR SUPERBODY
     1NO. , I4, 10H,  COLUMN , I4, 17H (   H   D.O.F. )    / )
   60 FORMAT  ( //6H   THE, I4, 40H  DHDX-ARRAY ELEMENTS FOR SUPERBODY
     1NO. , I4, 10H,  COLUMN , I4, 17H (   H   D.O.F. )    / )
   70 FORMAT  ( //    6H   THE, I4, 37H  H-ARRAY ELEMENTS FOR SUPERBODY
     1NO. , I4, 10H,  COLUMN , I4, 17H (ALPHA  D.O.F. )    / )
   80 FORMAT  ( //6H   THE, I4, 40H  DHDX-ARRAY ELEMENTS FOR SUPERBODY
     1NO. , I4, 10H,  COLUMN , I4, 17H ( ALPHA D.O.F. )    / )
C
      IF  (NRUN .EQ. 1)  REWIND  NTPSH
      IDOF       =  1
      ICOL       =  0
      DO   190   NX = 1, NXQ
C
   90 CONTINUE
      ICOL       =   ICOL + 1
      DO   100   I = 1, NSBETO
      H(I)       =  0.0
      DHDX(I)    =  0.0
  100 CONTINUE
      DO   160   J = 1, NSUP
      ISB        =   INSUP(J)
      I1         =  1
      IF   (ISB .EQ. 1)  GO TO  120
C
      I1         =  NBCUM(ISB-1) + 1
  120 I2         =  NBCUM(ISB)
      DO   150   I = I1, I2
      GO TO   (130, 140),   IDOF
  130 H(I)        =  1.0
      GO TO   150
C
  140 DHDX(I)   = -1.0
      H(I)       = -(X3L(I) - XIQ(NX))
  150 CONTINUE
C
  160 CONTINUE
      WRITE   (NTPSH)  ICOL, NSBETO,   ( H(I)   ,  I = 1, NSBETO)
      WRITE   (NTPSH)  ICOL, NSBETO,   (DHDX(I),  I = 1, NSBETO)
C
      IF  (IDOF .EQ. 2)  GO TO  180
      IDOF       =  2
C
```

```
      IF  (KPRINT .EQ. 0)  GO TO  90
      WRITE  (NTP6,50)  NSBETO, NRUN, ICOL
      WRITE  (NTP6,40)  ( H(I)   ,   I = 1, NSBETO)
      WRITE  (NTP6,60)  NSBETO, NRUN, ICOL
      WRITE  (NTP6,40)  (DHDX(I),   I = 1, NSBETO)
      GO TO  90
  180 CONTINUE
      IDOF      =  1
C
      IF  (KPRINT .EQ. 0)  GO TO  190
      WRITE  (NTP6,70)  NSBETO, NRUN, ICOL
      WRITE  (NTP6,40)  ( H(I)   ,   I = 1, NSBETO)
      WRITE  (NTP6,80)  NSBETO, NRUN, ICOL
      WRITE  (NTP6,40)  (DHDX(I),   I = 1, NSBETO)
  190 CONTINUE
C
      NMODE     =  NMODE + ICOL
      RETURN
      END
```

```
      SUBROUTINE NEWH(NTP5, NTP6, NTAPE, MASTH4, MASTH , MASTDH      , NEWH   2
     1               NTOT   , NSYM  , NASYM , NROW  , MODES , NBARAY,   NP , NEWH   3
     2               NCARAY , YS    , ZS    , Y     , X14C  , X34C        , NEWH   4
     3               H14C   , H34C  , HP    , NODE  , XI    , ETA         , NEWH   5
     4               ELXI   , ELYI  , ELZI  , HCOL  , PHI   , THB         , NEWH   6
     5               IFP    , IPAIR , I112  , SWORK         , WORK        , NEWH   7
     6               NODETO , NDOFTO, PHIT, MDOF, MNODE                   ) NEWH   8
C                                                                         NEWH   9
C         THIS SUBROUTINE IS THE MAIN PART OF THE  *NEW H-, ALPHA-,       NEWH  10
C         THETA-SPLINE*  PACKAGE                                          NEWH  11
C                                                                         NEWH  12
      DIMENSION  NBARAY(1), NCARAY(1), YS(1), ZS(1), Y(1),                NEWH  13
     1           X14C(1)   , X34C(1)   , H14C(1), H34C(1), HP(1)        , NEWH  14
     2           NODE(1)   , XI(1)     , ETA(1) , ELXI(1), ELYI(1)      , NEWH  15
     3           ELZI(1)   , HCOL(1)   , PHI(NDOFTO,MODES),  IFP(1)     , NEWH  16
     4           IPAIR(1)  , I112(2,   10,  10 ),         THB(1)        , NEWH  17
     5           SWORK(1),                       WORK(NTOT, NDOFTO)       NEWH  18
C                                                                         NEWH  19
      DIMENSION  INDX(200)                                                NEWH  20
      DIMENSION  PHIT(MDOF,MNODE,MODES)                                   NEWH  21
      DIMENSION  DOF(5), FIX(5), IHAT(5), XOAR(20), YOAR(20), NASS(50)    NEWH  22
      DIMENSION  XAR(50), YAR(50), ZAR(50), XBR(50), YBR(50), ZBR(50)     NEWH  23
      REAL       LMBDA                                                    NEWH  24
      DATA       DOF, FIX, KODE / 5*1H , 1HH, 1HA, 1HT, 1HB, 1HD, -1 /    NEWH  25
C                                                                         NEWH  26
   10 FORMAT   ( 6I12 )                                                   NEWH  27
   20 FORMAT   ( 5(11X, 1A1) )                                            NEWH  28
   30 FORMAT   ( 6F12.0 )                                                 NEWH  29
   50 FORMAT   ( 1H1/// 10X, 35H***    ENTER SUBROUTINE  NEWH    ***  ///  NEWH  30
     1           10X, 8HNSUPTO =, I4, 26H  (NUMBER OF SUPERPANELS)  / 10X, NEWH  31
     2           8HNODETO =, I4, 45H  (TOTAL NUMBER OF NODES IN ALL SUPERPANEWH  32
     3NELS) / 10X, 8HNDOFTO =, I4, 58H  (TOTAL NUMBER OF DEGREES-OF-FREENEWH  33
     4DOM IN ALL SUPERPANELS)  / 10X, 8HMAXSEC =, I4, 53H  (MAXIMUM NUMBNEWH  34
     5ER OF SECTIONS IN ANY ONE SUPERPANEL) / 10X, 8HMAXPR  =,  I4,     NEWH  35
     6 66H  (MAXIMUM NUMBER OF *FIRST-, LAST BOX PAIRS* IN ANY ONE SECTNEWH  36
     7ION)                                                      //    ) NEWH  37
   60 FORMAT   ( 1H1/// 10X, 10HSUPERPANEL , I3, 14H  INPUT VALUES  //    NEWH  38
     1           10X, 7HNSEC  =, I4, 22H  (NUMBER OF SECTIONS)    /       NEWH  39
     2           10X, 7HNODES =, I4, 22H  (NUMBER OF NODES)       /       NEWH  40
     3           10X, 7HDOF    =, 5(3X, A1), 22H  (DEGREES OF FREEDOM)  //  NEWH  41
     4           10X, 5HXA  =, F12.5, 7H  YA   =, F12.5, 7H  ZA   =, F12.5,  NEWH  42
     5            45H (INBOARD  EDGE COORD.-S OF ELASTIC AXIS)     / 10X,  NEWH  43
     6           5HXB  =, F12.5, 7H  YB  =, F12.5, 7H  ZB  =, F12.5    ,  NEWH  44
     7            45H (OUTBOARD EDGE COORD.-S OF ELASTIC AXIS)     //    )  NEWH  45
   70 FORMAT   (/// 10X, 11HSECTION NO. , I3, 15H  OF SUPERPANEL , I3 //  NEWH  46
     1           10X, 8HIFPRLL = , I3                          //  10X,   NEWH  47
     2           5HXO  =, F12.5, 7H  YO  =, F12.5, 7H  ZO  =, F12.5// 10X,  NEWH  48
     3           35HNUMBER OF FIRST-, LAST BOX PAIRS  = , I4   // 10X,    NEWH  49
     4           23HSET       FIRST        LAST / 20X, 3HBOX, 7X, 3HBOX   /  NEWH  50
     5           ( I12, 2I10 )                                          ) NEWH  51
   80 FORMAT   ( //    10X, 31HNODAL COORDINATES IN SUPERPANEL , I3   //  NEWH  52
     1           9X, 40HI         NODE          XI(NODE)       ETA(NODE)      / ) NEWH  53
```

```
   82 FORMAT   ( // 10X, 31HASSOCIATED NODES FOR SUPERPANEL , I3,
     1          24H ( TAKEN FROM SUPERPANEL , I3, 2H )// ( 8X, 12I6 ) )
   90 FORMAT   ( 2I10, 2F14.5 )
C
      DO   94   J = 1, MODES
      DO   92   I = 1, NDOFTO
      PHI(I,J)  =   0.0
   92 CONTINUE
   94 CONTINUE
      KX         =   0
      LNGTH      =   200
      IT         =   0
      CALL          OPENMS(NTAPE , INDX    , LNGTH    , IT        )
      LINES      =   45
C
      READ    (NTP5,10)   NSUPTO, MAXSEC, MAXPR
      WRITE   (NTP6,50)   NSUPTO, NODETO, NDOFTO, MAXSEC, MAXPR
C
      NDOF       =   0
      KN = 0
C
      DO   270   ISP = 1, NSUPTO
      READ    (NTP5,10)   NSEC, NODES, NEAASS, NODASS
      READ    (NTP5,20)   DOF
      READ    (NTP5,30)   XA, YA, ZA,  XB, YB, ZB
      WRITE   (NTP6,60)   ISP, NSEC, NODES, DOF, XA, YA, ZA, XB, YB, ZB
      XAR(ISP) =   XA
      YAR(ISP) =   YA
      ZAR(ISP) =   ZA
      XBR(ISP) =   XB
      YBR(ISP) =   YB
      ZBR(ISP) =   ZB
C
      DO   100   M = 1, 5
      IHAT(M)    =   0
      IF   (DOF(M) .EQ. FIX(M))   IHAT(M) = M
  100 CONTINUE
C
      XDIF       =   XB - XA
      YDIF       =   YB - YA
      ZDIF       =   ZB - ZA
      RHO        =   SQRT(YDIF**2 + ZDIF**2)
      COSG       =   YDIF / RHO
      SING       =   ZDIF / RHO
C
      IF   (NEAASS .NE. 0)   GO TO   120
      READ    (NTP5,10)   (NODE(I),  I = 1, NODES)
      WRITE   (NTP6,80)   ISP
      DO   110   I = 1, NODES
      ND         =   NODE(I)
      XI(I)      =   ELXI(ND)
      ETA(I)     =   (ELYI(ND)-YA)*COSG + (ELZI(ND)-ZA)*SING
```

```
      WRITE  (NTP6,90)  I, ND, XI(I), ETA(I)
      LX         =  ND
      DO   104  M = 1, 5
      IF   (IHAT(M) .EQ. 0)   GO TO  104
      KX         =   KX + 1
      DO   102  MX = 1, MODES
      MDF        =   3
      IF   (M .EQ. 2)   MDF = 5
      IF   (M .EQ. 3)   MDF = 4
      PHI(KX,MX)   =   PHIT(MDF, LX, MX)
      IF (M.EQ.3) PHI(KX,MX) = -PHI(KX,MX)
  102 CONTINUE
  104 CONTINUE
  110 CONTINUE
      GO TO  130
C
  120 CONTINUE
      IF   (NODES .NE. 0)  GO TO  130
      READ    (NTP5,10)   (NASS(I),   I = 1, NODASS)
      WRITE   (NTP6,82)   ISP, NEAASS, (NASS(I),   I = 1, NODASS)
      DO   124   ND = 1, NODASS
      NDAS       =   NASS(ND)
C
C          OBTAIN COORDINATES FOR THIS CONTROL SURFACE FROM ITS
C          ASSOCIATED NODE(S) ON THE ASSOCIATED ELASTIC AXIS   (NEAASS)
C
      CALL          CTLS(NEAASS, NDAS   , ND, ELXI, ELYI, ELZI, XA, YA, ZA,
     1              XB, YB, ZB, XAR, YAR, ZAR, XBR, YBR, ZBR,
     2              XI, ETA    ,                              NTP6            )
C
      LX         =   NDAS
      DO   134   M = 4, 5
      IF   (IHAT(M) .EQ. 0)   GO TO   134
      KX         =   KX + 1
      DO   132   MX = 1, MODES
      MDF    =   7
      IF   (M .EQ. 5)   MDF = 8
      PHI(KX, MX) =   PHIT(MDF, LX, MX)
  132 CONTINUE
  134 CONTINUE
  124 CONTINUE
      NODES      =   NODASS
      IHAT(2)    =   2
  130 CONTINUE
      YAB        =   0.0
      YBB        =   YDIF*COSG + ZDIF*SING
      CALL           ATAN3( XDIF, YBB , LMBDA )
C
  140 CONTINUE
C
C          GENERATE  *Y*-ARRAY FOR ALL BOXES  (FROM YS-  OR ZS-ARRAYS
C          FOR STRIPS  )
```

```
C
      J          =  1
      I2         =  0
C
      DO   170  L = 1, NP
      NC         =  NCARAY(L)
      NBCUM      =  NBARAY(L)
  150 CONTINUE
      I1         =  I2 + 1
      I2         =  I2 + NC
      DO   160  I = I1, I2
      Y(I)       =  (YS(J)-YA)*COSG +(ZS(J)-ZA)*SING
  160 CONTINUE
      J          =  J + 1
      IF  (I2 .GE. NBCUM)  GO TO  170
      GO TO  150
C
  170 CONTINUE
      NDIM       =  NODES + 4
      L1         =  1
      L2         =  L1 + NDIM * NDIM
      LENGTH     =  NDIM * (NDIM + NODES)
      CALL          ZEROUT(SWORK(L1),  LENGTH )
C
      CALL          SPLN(SWORK(L1), SWORK(L2), THB,          XI  , ETA    ,
     1              XA, YAB, XB, YBB,   NODES, NDIM , IPRINT,  NTP6     )
C
C          INPUT SECTION DATA  FOR SUPERPANEL
C
      DO  180  N = 1, NSEC
      READ (NTP5,10) IFP(N), IPAIR(N)
      READ (NTP5,30) XO, YO, ZO
      NPAIR      =  IPAIR(N)
      READ   (NTP5,10)  (I1I2(1,IPR,N), I1I2(2,IPR,N),   IPR = 1, NPAIR)
      WRITE  (NTP6,70)  N , ISP, IFP(N), XO, YO, ZO,  NPAIR,
     1              (IPR, I1I2(1,IPR,N), I1I2(2,IPR,N),  IPR = 1, NPAIR)
      XOAR(N)    =  XO
      YOAR(N)    =  (YO-YA)*COSG + (ZO-ZA)*SING
C
      IF  (N .LT. NSEC)  WRITE  (NTP6,240)
  180 CONTINUE
C
C          BEGIN DO LOOP ON ALL NODES OF SUPERPANEL  *ISP*
C
      LCUM       =  0
      LIMIT      =  LINES - 5
      IF  (IPRINT .EQ. 2)  WRITE  (NTP6,240)
      DO   260  L = 1, NODES
      K          =  L2 + (L-1) * NDIM
      NTH        =  NODES + 2
      DO   250  M = 1, 3
      IGO        =  IHAT(M)
```

```
      IF   (IGO .EQ. 0)  GO TO  250
      NDOF        =  NDOF + 1
      DO  190   I = 1, NTOT
      H14C(I)     =  0.0
      H34C(I)     =  0.0
  190 HP(I)       =  0.0
C
C          BEGIN DO LOOP ON ALL SECTIONS OF SUPERPANEL   *ISP*
C
      DO   200   N = 1, NSEC
      XO          =  XOAR(N)
      YO          =  YOAR(N)
C
      CALL          HATS( XA , YA , XO , YO , LMBDA , X14C , X34C , Y    ,
     1              H14C, H34C, HP,       M   , IFP(N), NPAIR, I1I2(1,1,N),
     2              NTH , SWORK(K),            THR                        )
C
  200 CONTINUE
C
C          END DO  LOOP ON *M*  (1 FOR H,  2 FOR ALPHA,  3 FOR THETA)
C
      KN1         =  KN + 1
      KN2         =  KN + 2
      KN3         =  KN + 3
      CALL          WRITMS(NTAPE , H14C   , NTOT    , KN1        )
      CALL          WRITMS(NTAPE , H34C   , NTOT    , KN2        )
      CALL          WRITMS(NTAPE ,  HP    , NTOT    , KN3        )
C
      KN          =  KN3
      IF   (IPRINT .NE. 2)  GO TO  250
      WRITE  (NTP6,220)  DOF(M),  L
      LCUM        =  LCUM + 5
C
      DO  210  I = 1, NTOT
      LCUM        =  LCUM + 1
      WRITE   (NTP6,230)  I, H14C(I), H34C(I), HP(I)
      IF   (LCUM .LT. LIMIT .OR. I .EQ. NTOT)  GO TO  210
      LIMIT       =  LIMIT + LINES
      LCUM        =  LCUM + 5
      WRITE  (NTP6,240)
      WRITE  (NTP6,220)   DOF(M),  L
  210 CONTINUE
C
  220 FORMAT    (      // 10X, 31HDEFLECTIONS AND SLOPES DUE TO   , 1A1 ,
     1           17H  OF NODE NUMBER , I3  // 12X, 3HBOX, 9X, 7HH-1/4-C ,
     2           7X, 7HH-3/4-C  , 8X, 6HDH/DX                    //    )
  230 FORMAT    ( I15, 3F14.5  )
  240 FORMAT    ( 1H1 //  )
C
  250 CONTINUE
  260 CONTINUE
  270 CONTINUE
```

```
C
C          BEGIN MATRIX MULTIPLICATION  TO OBTAIN  *H* AND *HP* COLUMNS
C          PER MODE  (PHI-COLUMNS ARE EXPECTED TO BE  *NDOFTO*  LONG)
C
      DO   420  IGO = 1, 3
      GO TO  (280,290,300),   IGO
  280 NTHOUT    =   MASTH4
      GO TO  310
  290 NTHOUT    =   MASTH
      GO TO  310
  300 NTHOUT    =   MASTDH
  310 CONTINUE
C
      WRITE  (NTHOUT)   NSYM, NASYM
      KN        =   IGO - 3
C
      DO   320  J = 1, NDOF
      KN        =   KN + 3
      CALL          READMS(NTAPE  , WORK(1,J), NTOT   , KN        )
C     WRITE  (NTP6,440)  (WORK(I,J),  I = 1, NTOT)
  320 CONTINUE
C
      M1        =   1
      M2        =   NSYM
      IF   (IPRINT .NE. 0)  WRITE  (NTP6,470)
      LOOP      =   1
      LIMIT     =   LINES - 5
      LCUM      =   0
  330 CONTINUE
      MODE      =   0
      DO   400  M = M1, M2
C     WRITE  (NTP6,470)  M
C     WRITE  (NTP6,440)  (PHI(J,M),  J = 1, NDOFTO)
C
      MODE      =   MODE + 1
      DO   350  I = 1, NTOT
      HCOL(I)   =   0.0
C
      DO   340  J = 1, NDOF
      HCOL(I)   =   HCOL(I) + WORK(I,J) * PHI(J, M)
  340 CONTINUE
  350 CONTINUE
C
      MH        =   M - M1 + 1
      WRITE  (NTHOUT)  MH
      WRITE  (NTHOUT)  (HCOL(I),  I = 1, NTOT)
      IF  (IPRINT .EQ. 0)  GO TO  400
      GO TO  (360,370,380),  IGO
  360 WRITE  (NTP6,430)  MODE
      GO TO  390
  370 WRITE  (NTP6,440)  MODE
      GO TO  390
```

```
  380 WRITE  (NTP6,450)   MODE                                                    NEWH 314
  390 CONTINUE                                                                    NEWH 315
      WRITE  (NTP6,460)   (HCOL(I),  I = 1, NTOT)                                 NEWH 316
      LCUM      =   LCUM + NTOT / 6                                               NEWH 317
      IF   (LCUM .LT. LIMIT .OR. M .EQ. M2)   GO TO  400                          NEWH 318
      IF   (LOOP .EQ. 1)   WRITE  (NTP6,470)                                      NEWH 319
      LCUM      =   LCUM + 5                                                      NEWH 320
      LIMIT     =   LIMIT + LINES                                                 NEWH 321
  400 CONTINUE                                                                    NEWH 322
C                                                                                 NEWH 323
      WRITE  (NTHOUT)   KODE                                                      NEWH 324
      IF   (NASYM .EQ. 0 .OR. LOOP .EQ. 2)   GO TO  410                           NEWH 325
      LOOP      =   2                                                             NEWH 326
      M1        =   NSYM + 1                                                      NEWH 327
      M2        =   NSYM + NASYM                                                  NEWH 328
      IF   (IPRINT .NE. 0)   WRITE  (NTP6,480)                                    NEWH 329
      GO TO  330                                                                  NEWH 330
C                                                                                 NEWH 331
  410 CONTINUE                                                                    NEWH 332
C                                                                                 NEWH 333
  420 CONTINUE                                                                    NEWH 334
C                                                                                 NEWH 335
  430 FORMAT  ( // 10X, 29HH-1/4-C MATRIX COLUMN  (MODE) , I3    /     ) NEWH 336
  440 FORMAT  ( // 10X, 29HH-3/4-C MATRIX COLUMN  (MODE) , I3    /     ) NEWH 337
  450 FORMAT  ( // 10X, 29HDH / DX MATRIX COLUMN  (MODE) , I3    /     ) NEWH 338
  460 FORMAT  ( 5X, 6E15.6 )                                               NEWH 339
  470 FORMAT  ( 1H1 /// 10X, 32H***     SYMMETRIC MODES     ***   /     ) NEWH 340
  480 FORMAT  (       // 10X, 32H***   ANTISYMMETRIC MODES   ***   /     ) NEWH 341
C                                                                                 NEWH 342
      CALL        CLOSMS( NTAPE )                                                 NEWH 343
      REWIND   MASTH4                                                             NEWH 344
      REWIND   MASTH                                                              NEWH 345
      REWIND   MASTDH                                                             NEWH 346
      RETURN                                                                      NEWH 347
      END                                                                         NEWH 348
```

```
      SUBROUTINE ORGN (NTAPE,MASTAP,NSYM,NASYM,NODES,MODES,IROW,
     1 COL,PHI,WORK )
C
      DIMENSION PHI(NODES,MODES), COL(IROW), WORK(IROW,1)
C
      DATA        KODE  /  -1  /,  NTSCR  /  8  /
C
      IGO        =  1
      M1         =  1
      M2         =  NSYM
      NCOL       =  NSYM
      REWIND  NTSCR
C
      WRITE  (MASTAP)  NSYM, NASYM
C
    7 CONTINUE
      DO  50  JC = 1, NCOL
      DO  9  JR = 1, IROW
    9 WORK(JR,JC) =  0.0
   50 CONTINUE
   10 CONTINUE
      READ   (NTAPE)  NODE
      IF  (IGO .EQ. 1)  WRITE  (NTSCR)  NODE
      IF  (NODE .EQ. KODE)  GO TO  40
      READ   (NTAPE)  NTOT, (COL(I),  I = 1, NTOT)
      IF  (IGO .EQ. 1)  WRITE  (NTSCR)  NTOT, (COL(I),  I = 1, NTOT)
      DO  30  MD = M1, M2
      MH         =  MD - M1 + 1
      DO  20  IR = 1, NTOT
      WORK(IR,MH)  =  WORK(IR,MH) + COL(IR)*PHI(NODE, MD)
   20 CONTINUE
C
   30 CONTINUE
      GO TO  10
C
   40 CONTINUE
C
      DO  80  JC =     1, NCOL
      WRITE  (MASTAP)  JC
      WRITE  (MASTAP)  (WORK(I, JC),  I =  1, NTOT)
   80 CONTINUE
   90 CONTINUE
      WRITE  (MASTAP)  KODE
C
      IF  (IGO .EQ. 2 .OR. NASYM .EQ. 0)  GO TO  100
      IGO        =  2
      M1         =  M2 + 1
      M2         =  MODES
      NCOL       =  NASYM
      NTOLD      =  NTAPE
      NTAPE      =  NTSCR
      REWIND  NTAPE
```

```
      GO TO  7                                                         ORGN  54
C                                                                      ORGN  55
  100 NTAPE     =  NTOLD                                               ORGN  56
      RETURN                                                           ORGN  57
      END                                                              ORGN  58
```

```
      SUBROUTINE PISTON(  NB    , NBOX   , NSBETO, KR     , CBAR   , FMACH, PISTON 2
     1                  NTPH   , NTPDH , NTPH4 , MASTCP, MASTFZ, MASTFY,NEWTPH, PISTON 3
     2                  NBARAY, NCARAY, NSBEA , YB     , ZB     , AO     , AOP   , PISTON 4
     3                  XIS1   , XIS2   , CG, SG, EE, YS, ZS, H , DHDX   , HP    , PISTON 5
     4                  XC     ,                          JRUN   , IPRINT        , PISTON 6
     5                  DCP    , DCPS   , DCPA   , DFQ    , DFQS   , DFQA   , NR,CR) PISTON 7
C                                                                         PISTON 8
C          THIS SUBROUTINE COMPUTES PRESSURES AND BODY FORCES             PISTON 9
C          DUE TO MODES   AND   GUSTS   USING  --                         PISTON10
C                                ***   PISTON THEORY   ***                PISTON11
C                                                                         PISTON12
      DIMENSION   NBARAY(1), NCARAY(1), NSBEA(1), YB(1), ZB(1), AO(1)   , PISTON13
     1            AOP(1),      XIS1(1)   , XIS2(1) , CG(1), SG(1), EE(1)   , PISTON14
     2            YS(1), ZS(1), XC(4,350)                                  PISTON15
      DIMENSION   H( 1 ), DHDX( 1 ), HP( 1 ), CR(3,20)                    PISTON16
      COMPLEX     DCP(1), DCPS(1), DCPA(1), DFQ(1), DFQS(1), DFQA(1)      PISTON17
     1     ,      DCP1, DCP2, DFQ1, DFQ2, E1AK, DCPTRM, TERM2, FPART      PISTON18
      COMPLEX      EILNOT                                                 PISTON19
      REAL        KR,  KBJ,  LMBNOT                                       PISTON20
      DATA        PI / 3.1415926 /, NTP6 / 6 /                            PISTON21
      DATA        KODE, NTEMP  / -1,  10 /                                PISTON22
C                                                                         PISTON23
   10 FORMAT  ( 1H1 /// 10X, 32HENTER SUBROUTINE  PISTON          /    )  PISTON24
      WRITE   (NTP6,10)                                                   PISTON25
C                                                                         PISTON26
      TWOPOM    =   2.0*PI / FMACH                                        PISTON27
      FOUROM    =   4.0 / FMACH                                           PISTON28
      TWOM      =   2.0 * FMACH                                           PISTON29
      NSLEND    =   NSBETO + NB                                           PISTON30
C                                                                         PISTON31
      REWIND  MASTCP                                                      PISTON32
      REWIND  NTEMP                                                       PISTON33
      IF  (JRUN .EQ. 1)  REWIND  NEWTPH                                   PISTON34
      REWIND  NTPH                                                        PISTON35
      REWIND  NTPDH                                                       PISTON36
      REWIND  NTPH4                                                       PISTON37
      READ    (NTPH )  NSYM, NASYM                                        PISTON38
      READ    (NTPDH)                                                     PISTON39
      IF  (NB .EQ. 0)  GO TO  120                                         PISTON40
C                                                                         PISTON41
      REWIND  MASTFZ                                                      PISTON42
      REWIND  MASTFY                                                      PISTON43
      NFTAPE    =   MASTFZ                                                PISTON44
      NMTO      =   NSYM                                                  PISTON45
      DO  110   IZY = 1, 4                                                PISTON46
      DO  100   NM  = 1, NMTO                                             PISTON47
      READ    (NTPH )  ID                                                 PISTON48
      READ    (NTPDH)                                                     PISTON49
      READ    (NTPH )  ( H  (I),   I = 1, NSLEND)                         PISTON50
      READ    (NTPDH)  (DHDX(I),   I = 1, NSLEND)                         PISTON51
      LJ2       =   0                                                     PISTON52
      IH        =   0                                                     PISTON53
```

```
C                                                                               PISTON54
      DO  30  LB = 1, NB                                                        PISTON55
      LJ1        =  LJ2 + 1                                                     PISTON56
      LJ2        =  LJ2 + NSBEA(LB)                                             PISTON57
C                                                                               PISTON58
      DO  20   LJ = LJ1, LJ2                                                    PISTON59
      IH         =  IH + 1                                                      PISTON60
      HB         =  0.5 * ( H   (LJ)  +  H   (LJ+1))                            PISTON61
      HBP        =  0.5 * (DHDX(LJ)  + DHDX(LJ+1))                              PISTON62
C                                                                               PISTON63
      TWOAO      =  2.0*AO(IH)                                                  PISTON64
      AB         =  TWOPOM *((AOP(IH) - 1.0/TWOM) * HB + TWOAO * HBP)           PISTON65
      BB         =  TWOPOM * TWOAO * HB / CBAR                                  PISTON66
C                                                                               PISTON67
      DELX       =  XIS2(IH) - XIS1(IH)                                         PISTON68
      DFQ(IH)    =-(CMPLX( AB, KR*BB )) * DELX                                  PISTON69
C                                                                               PISTON70
   20 CONTINUE                                                                  PISTON71
C                                                                               PISTON72
      LJ2        =  LJ2 + 1                                                     PISTON73
   30 CONTINUE                                                                  PISTON74
      IF   (IPRINT .LT. 2)  GO TO   90                                          PISTON75
      GO TO   (40,50,60,70),   IZY                                              PISTON76
   40 WRITE   (NTP6,440)  NM                                                    PISTON77
      GO TO   80                                                                PISTON78
   50 WRITE   (NTP6,450)  NM                                                    PISTON79
      GO TO   80                                                                PISTON80
   60 WRITE   (NTP6,460)  NM                                                    PISTON81
      GO TO   80                                                                PISTON82
   70 WRITE   (NTP6,470)  NM                                                    PISTON83
   80 CONTINUE                                                                  PISTON84
C                                                                               PISTON85
      WRITE   (NTP6,480)  (DFQ(I),  I = 1, NSBETO)                              PISTON86
   90 CONTINUE                                                                  PISTON87
      WRITE   (NFTAPE)    (DFQ(I),  I = 1, NSBETO)                              PISTON88
C                                                                               PISTON89
  100 CONTINUE                                                                  PISTON90
C                                                                               PISTON91
C          READ THE CODE  -1     ( END INDICATOR OF  H-, DHDX- MATRICES)        PISTON92
      READ    (NTPH )                                                           PISTON93
      READ    (NTPDH)                                                           PISTON94
C                                                                               PISTON95
      NMTO       =  NASYM                                                       PISTON96
      NFTAPE     =  NTEMP                                                       PISTON97
      IF   (IZY .NE. 2)  GO TO  110                                             PISTON98
C          READ *HEADER*  OF NEXT  H-, DHDX- MATRIX                             PISTON99
      READ    (NTPH )  NSYM, NASYM                                              PISTO100
      READ    (NTPDH)                                                           PISTO101
      NMTO       =  NSYM                                                        PISTO102
      NFTAPE     =  MASTFY                                                      PISTO103
C                                                                               PISTO104
  110 CONTINUE                                                                  PISTO105
```

```
C                                                                       PISTO106
C         BODY FORCES DUE TO SYMMETRIC MODES ARE SAVED ON TAPES         PISTO107
C         MASTFZ  AND MASTFY,  WHILE FORCES DUE TO ANTISYMMETRIC MODES  PISTO108
C         ARE SAVED ON TAPE  NTEMP                                      PISTO109
C         NEXT,  COMPUTE PRESSURES DUE TO MODES  ON LIFTING SURFACE     PISTO110
C         BOXES                                                         PISTO111
C                                                                       PISTO112
      READ    (NTPH )  NSYM, NASYM                                      PISTO113
      READ    (NTPDH)                                                   PISTO114
  120 READ    (NTPH4)                                                   PISTO115
C                                                                       PISTO116
      IF  (JRUN .EQ. 1)  WRITE  (NEWTPH)  NSYM, NASYM                   PISTO117
      NTAPE      =  MASTCP                                              PISTO118
      NMTO       =  NSYM                                                PISTO119
C                                                                       PISTO120
      DO  190  JM = 1, 2                                                PISTO121
      DO  180  NM = 1, NMTO                                             PISTO122
      READ    (NTPH )       ID                                          PISTO123
      READ    (NTPH )  ( H  (I),   I = 1, NBOX)                         PISTO124
      READ    (NTPDH)                                                   PISTO125
      READ    (NTPDH)  (DHDX(I),   I = 1, NBOX)                         PISTO126
      READ    (NTPH4)                                                   PISTO127
      READ    (NTPH4)  ( HP (I),   I = 1, NBOX)                         PISTO128
C                                                                       PISTO129
      DO  130  I = 1, NBOX                                              PISTO130
      IF  (JRUN .GT. 1)  GO TO 124                                      PISTO131
      HP(I)      = 0.5 * (H(I) + HP(I))                                 PISTO132
  124 CONTINUE                                                          PISTO133
      HPP        =  DHDX(I)                                             PISTO134
      AC         =  FOUROM * HPP                                        PISTO135
      BC         =  FOUROM * 2.0 * HP(I) / CBAR                         PISTO136
C                                                                       PISTO137
      DCP(I)     = -CMPLX( AC, KR*BC )                                  PISTO138
  130 CONTINUE                                                          PISTO139
      IF  (IPRINT .LT. 2)  GO TO  170                                   PISTO140
C                                                                       PISTO141
      GO TO   (140,150),  JM                                            PISTO142
  140 WRITE   (NTP6,490)   NM                                           PISTO143
      GO TO   160                                                       PISTO144
  150 WRITE   (NTP6,500)   NM                                           PISTO145
C                                                                       PISTO146
  160 CONTINUE                                                          PISTO147
C                                                                       PISTO148
      WRITE   (NTP6,480)   (DCP(I), I = 1, NBOX)                        PISTO149
  170 CONTINUE                                                          PISTO150
      WRITE   (NTAPE)      (DCP(I), I = 1, NBOX)                        PISTO151
      IF  (JRUN .GT. 1)    GO TO  180                                   PISTO152
      WRITE   (NEWTPH)     NM                                           PISTO153
      WRITE   (NEWTPH)     ( HP(I), I = 1, NBOX)                        PISTO154
C                                                                       PISTO155
  180 CONTINUE                                                          PISTO156
C                                                                       PISTO157
```

```
C         READ THE CODE  -1      ( END INDICATOR OF  H-, DHDX- MATRICES) PISTO158
C                                                                         PISTO159
      READ   (NTPH )                                                      PISTO160
      READ   (NTPDH)                                                      PISTO161
      READ   (NTPH4)                                                      PISTO162
      IF  (JRUN .EQ. 1)  WRITE  (NEWTPH)  KODE                            PISTO163
C                                                                         PISTO164
      NTAPE     =  NTEMP                                                  PISTO165
      NMTO      =  NASYM                                                  PISTO166
  190 CONTINUE                                                            PISTO167
C                                                                         PISTO168
C         PRESSURES DUE TO MODES COMPLETED  --  BEGIN CALCULATIONS  DUE  PISTO169
C         TO  GUSTS                                                       PISTO170
C                                                                         PISTO171
      DO  320  IR = 1, NR                                                 PISTO172
      COSA      =  CR(1,  IR)                                             PISTO173
      COSB      =  CR(2,  IR)                                             PISTO174
      COSC      =  CR(3,  IR)                                             PISTO175
C                                                                         PISTO176
      R         =  FMACH / (1.0 + FMACH*COSA)                             PISTO177
      TAU       =  2.0*KR * R / CBAR                                      PISTO178
      LP        =  1                                                      PISTO179
      IS        =  1                                                      PISTO180
      NC        =  NCARAY(LP)                                             PISTO181
      NBXCUM    =  NC                                                     PISTO182
C                                                                         PISTO183
      DO  300  I = 1, NBOX                                                PISTO184
      DXIB      =  0.5*(XC(2,I)+XC(4,I) - XC(1,I)-XC(3,I))                PISTO185
      IGO       =  1                                                      PISTO186
      THETA     =  -SG(IS)*COSB + CG(IS)*COSC                             PISTO187
      XLI       =  XC(1,  I)                                              PISTO188
      XTI       =  XC(2,  I)                                              PISTO189
      XLO       =  XC(3,  I)                                              PISTO190
      XTO       =  XC(4,  I)                                              PISTO191
      G         =  CG(IS)*COSB + SG(IS)*COSC                              PISTO192
C                                                                         PISTO193
      IF  (ABS(COSA) .LE. 0.001)  GO TO 270                               PISTO194
      YMULT     =  YS(IS) - EE(IS) * CG(IS)                               PISTO195
      ZMULT     =  ZS(IS) - EE(IS) * SG(IS)                               PISTO196
      DLTRM     =  2.0*EE(IS) * (CG(IS)*COSB + SG(IS)*COSC)               PISTO197
  200 ADDON     =  YMULT*COSB + ZMULT*COSC                                PISTO198
C                                                                         PISTO199
      ELM1      =  XLI * COSA + ADDON                                     PISTO200
      ELM2      =  XTI * COSA + ADDON                                     PISTO201
      ARG31     =  TAU * ((XLO - XLI) * COSA + DLTRM)                     PISTO202
      ARG42     =  TAU * ((XTO - XTI) * COSA + DLTRM)                     PISTO203
C                                                                         PISTO204
      IF  (ABS(ARG31) .LE. 0.01)  GO TO  210                              PISTO205
      A31       =  (COS(ARG31) - 1.0) / ARG31                             PISTO206
      B31       =   SIN(ARG31) / ARG31                                    PISTO207
      GO TO  220                                                          PISTO208
C                                                                         PISTO209
```

```
  210 CONTINUE                                                                     PIST0210
      A31         =   -ARG31 / 2.0                                                 PIST0211
      B31         =   1.0                                                          PIST0212
  220 CONTINUE                                                                     PIST0213
C                                                                                  PIST0214
      IF  (ABS(ARG42) .LE. 0.01)  GO TO  230                                       PIST0215
      A42         =   (COS(ARG42) - 1.0) / ARG42                                   PIST0216
      B42         =    SIN(ARG42) / ARG42                                          PIST0217
      GO TO  240                                                                   PIST0218
  230 CONTINUE                                                                     PIST0219
      A42         =   -ARG42 / 2.0                                                 PIST0220
      B42         =   1.0                                                          PIST0221
C                                                                                  PIST0222
  240 CONTINUE                                                                     PIST0223
C                                                                                  PIST0224
      COSTL1      =   COS( TAU*ELM1)                                               PIST0225
      COSTL2      =   COS( TAU*ELM2)                                               PIST0226
      SINTL1      =   SIN( TAU*ELM1)                                               PIST0227
      SINTL2      =   SIN( TAU*ELM2)                                               PIST0228
C                                                                                  PIST0229
      DCPTRM      =   CMPLX(( A31*COSTL1 - B31*SINTL1                              PIST0230
     1                      -  A42*COSTL2 + B42*SINTL2)      ,                     PIST0231
     2                (-B31*COSTL1 - A31*SINTL1 + B42*COSTL2 + A42*SINTL2))        PIST0232
C                                                                                  PIST0233
      DENCM       =   DXIB * TAU * COSA                                            PIST0234
C                                                                                  PIST0235
      GO TO  (250,260),  IGO                                                       PIST0236
  250 CONTINUE                                                                     PIST0237
      DCP1        =   THETA * FOUROM * DCPTRM / DENOM                              PIST0238
C                                                                                  PIST0239
      IGO         =   2                                                            PIST0240
      THETA       =   SG(IS)*COSB + CG(IS)*COSC                                    PIST0241
      YMULT       = -YS(IS) - EE(IS)*CG(IS)                                        PIST0242
      ZMULT       =   ZS(IS) + EE(IS)*SG(IS)                                       PIST0243
      DLTRM       =   2.0*EE(IS) * (CG(IS)*COSB - SG(IS)*COSC)                     PIST0244
      XLI         =   XC(3, I)                                                     PIST0245
      XTI         =   XC(4, I)                                                     PIST0246
      XLO         =   XC(1, I)                                                     PIST0247
      XTO         =   XC(2, I)                                                     PIST0248
      GO TO  200                                                                   PIST0249
C                                                                                  PIST0250
  260 CONTINUE                                                                     PIST0251
C                                                                                  PIST0252
      DCP2        =   THETA * FOUROM * DCPTRM / DENOM                              PIST0253
      GO TO  280                                                                   PIST0254
  270 CONTINUE                                                                     PIST0255
      LMBNOT      =   TAU * (YS(IS)*COSB + ZS(IS)*COSC)                            PIST0256
  272 TEG         =   TAU * EE(IS) * G                                             PIST0257
      IF  (ABS(TEG) . LE. 0.001)  GO TO  278                                       PIST0258
      Q           =   SIN(TEG) / TEG                                               PIST0259
      TEGMLT      =   (COS(TEG) - Q) / TEG                                         PIST0260
  274 CONTINUE                                                                     PIST0261
```

```
      EILNOT    =  CMPLX(COS(LMBNOT), -SIN(LMBNOT))                        PIST0262
      DCPTRM    =  CMPLX( Q , ( (XLI-XTI-XLO+XTO) / (XLI-XTI+XLO-XTO) )    PIST0263
     1             * TEGMLT )                                              PIST0264
C                                                                          PIST0265
      GO TO  (276, 279),  IGO                                              PIST0266
  276 CONTINUE                                                             PIST0267
      DCP1      =   THETA * FOUROM * EILNOT * DCPTRM                       PIST0268
      IGO       =  2                                                       PIST0269
      THETA     =  SG(IS)*COSB + CG(IS)*COSC                               PIST0270
      G         =  CG(IS)*COSB - SG(IS)*COSC                               PIST0271
      LMBNOT    =  TAU * (-YS(IS)*COSB + ZS(IS)*COSC)                      PIST0272
      XLI       =  XC(3, I)                                                PIST0273
      XTI       =  XC(4, I)                                                PIST0274
      XLO       =  XC(1, I)                                                PIST0275
      XTO       =  XC(2, I)                                                PIST0276
      GO TO  272                                                           PIST0277
  278 Q         =  1.0                                                     PIST0278
      TEGMLT    =  0.0                                                     PIST0279
      GO TO  274                                                           PIST0280
  279 CONTINUE                                                             PIST0281
      DCP2      =   THETA * FOUROM * EILNOT * DCPTRM                       PIST0282
C                                                                          PIST0283
  280 CONTINUE                                                             PIST0284
      DCPS(I)   =  0.5 * (DCP1 + DCP2)                                     PIST0285
      DCPA(I)   =  0.5 * (DCP1 - DCP2)                                     PIST0286
C                                                                          PIST0287
      IF  (I .LT. NBXCUM)        GO TO  300                                PIST0288
      IF  (I .NE. NBARAY(LP))    GO TO  290                                PIST0289
      LP        =  LP + 1                                                  PIST0290
      NC        =  NCARAY(LP)                                              PIST0291
  290 IS        =  IS + 1                                                  PIST0292
      NBXCUM    =  NBXCUM + NC                                             PIST0293
C                                                                          PIST0294
  300 CONTINUE                                                             PIST0295
C                                                                          PIST0296
C          ONE SYMMETRIC, AND ONE ANTISYMMETRIC PRESSURE COLUMN DUE TO     PIST0297
C          GUST  COMPLETE,  SAVE ON TAPE  (AND PRINT)                      PIST0298
C                                                                          PIST0299
      IF  (IPRINT .LT. 2)  GO TO  310                                      PIST0300
      WRITE  (NTP6,510)  IR                                                PIST0301
      WRITE  (NTP6,480)  (DCPS(I),  I = 1, NBOX)                           PIST0302
      WRITE  (NTP6,520)  IR                                                PIST0303
      WRITE  (NTP6,480)  (DCPA(I),  I = 1, NBOX)                           PIST0304
  310 CONTINUE                                                             PIST0305
      WRITE  (MASTCP)    (DCPS(I),  I = 1, NBOX)                           PIST0306
      WRITE  (NTEMP)     (DCPA(I),  I = 1, NBOX)                           PIST0307
  320 CONTINUE                                                             PIST0308
C                                                                          PIST0309
C          PRESSURES DUE TO GUSTS COMPLETE --- COMPUTE BODY FORCES         PIST0310
C          DUE TO GUSTS                                                    PIST0311
C          THE VARIABLES  DCP1, DCP2, DCPS, DCPA  ARE USED HERE FOR THE    PIST0312
C          TEMPORARY STORAGE OF THE  Y-FORCE COMPONENTS                    PIST0313
```

```
C                                                                       PIST0314
      DO  430   IR = 1, NR                                              PIST0315
      COSA      =   CR(1, IR)                                           PIST0316
      COSB      =   CR(2, IR)                                           PIST0317
      COSC      =   CR(3, IR)                                           PIST0318
      R         =   FMACH / (1.0 + FMACH*COSA)                          PIST0319
      LB2       =   0                                                   PIST0320
C                                                                       PIST0321
      DO  410   LB = 1, NB                                              PIST0322
      THETAZ    =   COSC                                                PIST0323
C                                                                       PIST0324
      LB1       =   LB2 + 1                                             PIST0325
      LB2       =   LB2 + NSBEA(LB)                                     PIST0326
      DO 400   I =  LB1, LB2                                            PIST0327
      IZY       =   1                                                   PIST0328
      THETAY    =   COSB                                                PIST0329
      IF  (YB(LB)  .EQ. 0.0)   GO TO  330                               PIST0330
      THETAY    =   YB(LB)*COSB / ABS(YB(LB))                           PIST0331
  330 CONTINUE                                                          PIST0332
C                                                                       PIST0333
      YTRM      =   YB(LB) * COSB                                       PIST0334
      ZTRM      =   ZB(LB) * COSC                                       PIST0335
      ADDON     =   YTRM + ZTRM                                         PIST0336
      DELX      =   XIS2(I) - XIS1(I)                                   PIST0337
      XO        =   0.5 * (XIS1(I) + XIS2(I))                           PIST0338
      BJ        =   DELX * R * COSA / CBAR                              PIST0339
      KBJ       =   KR * BJ                                             PIST0340
      C1        =   2.0*PI * AO(I) * (1.0 - R*COSA) / FMACH             PIST0341
      C2        =   -PI * CBAR * (AOP(I) - 0.5 / FMACH) / FMACH         PIST0342
      C2OKR     =   0.0                                                 PIST0343
      IF  (KR .NE. 0.0)  C2OKR = C2 / KR                                PIST0344
      TERM1     =   1.0                                                 PIST0345
      IF  (ABS(KBJ) .LE. 0.001)  GO TO  340                             PIST0346
      TERM1     =   SIN(KBJ) / KBJ                                      PIST0347
  340 CONTINUE                                                          PIST0348
C                                                                       PIST0349
  350 CONTINUE                                                          PIST0350
      AJ        =   2.0 * R * (XO * COSA + ADDON) / CBAR                PIST0351
      AJK       =   AJ * KR                                             PIST0352
      E1AK      =   CMPLX( COS(AJK), -SIN(AJK) )                        PIST0353
      TERM2     =   CMPLX ( C1, C2OKR )                                 PIST0354
      FPART     =   E1AK * TERM1 * TERM2                                PIST0355
      GO TO   (360,380),   IZY                                          PIST0356
  360 DFQ1      =   THETAZ * FPART                                      PIST0357
      DCP1      =   THETAY * FPART                                      PIST0358
C                                                                       PIST0359
      IF  (YB(LB)  .EQ. 0.0)   GO TO  370                               PIST0360
      YTRM      =   -YB(LB) * COSB                                      PIST0361
      ADDON     =   YTRM + ZTRM                                         PIST0362
      THETAY    =   YTRM / ABS(YB(LB))                                  PIST0363
      IZY       =   2                                                   PIST0364
      GO TO  350                                                        PIST0365
```

```
C
  370 DFQ2      =  DFQ1
      DCP2      =  DCP1
      GO TO  390
C
  380 CONTINUE
      DFQ2      =  THETAZ * FPART
      DCP2      =  THETAY * FPART
C
  390 DFQS(I)   =  0.5 * (DFQ1 + DFQ2) * DELX
      DFQA(I)   =  0.5 * (DFQ1 - DFQ2) * DELX
      DCPS(I)   =  0.5 * (DCP1 + DCP2) * DELX
      DCPA(I)   =  0.5 * (DCP1 - DCP2) * DELX
C
  400 CONTINUE
  410 CONTINUE
C         ONE SYMMETRIC, AND ONE ANTISYMMETRIC BODY FORCE COLUMN  DUE
C         TO GUSTS COMPLETE   --   WRITE ON TAPE   (AND PRINT)
C
      IF  (IPRINT .LT. 2)  GO TO  420
      WRITE  (NTP6,530)  IR
      WRITE  (NTP6,480)  (DFQS(I),  I = 1, NSBETO)
      WRITE  (NTP6,540)  IR
      WRITE  (NTP6,480)  (DFQA(I),  I = 1, NSBETO)
      WRITE  (NTP6,550)  IR
      WRITE  (NTP6,480)  (DCPS(I),  I = 1, NSBETO)
      WRITE  (NTP6,560)  IR
      WRITE  (NTP6,480)  (DCPA(I),  I = 1, NSBETO)
  420 CONTINUE
      WRITE  (MASTFZ)    (DFQS(I),  I = 1, NSBETO)
      WRITE  (MASTFY)    (DCPS(I),  I = 1, NSBETO)
      WRITE  (NTEMP )    (DFQA(I),  I = 1, NSBETO)
      WRITE  (NTEMP )    (DCPA(I),  I = 1, NSBETO)
  430 CONTINUE
C
  440 FORMAT  (    // 10X, 36HZ-BODY FORCES DUE TO SYMMETRIC MODE ,I4/)
  450 FORMAT  (    // 10X, 36HZ-BODY FORCES DUE TO ANTISYMM. MODE ,I4/)
  460 FORMAT  (    // 10X, 36HY-BODY FORCES DUE TO SYMMETRIC MODE ,I4/)
  470 FORMAT  (    // 10X, 36HY-BODY FORCES DUF TO ANTISYMM. MODE ,I4/)
  480 FORMAT  ( 6E16.6 )
  490 FORMAT  (    // 10X, 36HPRESSURES DUE TO SYMMETRIC  MODE     ,I4/)
  500 FORMAT  (    // 10X, 36HPRESSURES DUE TO ANTISYMM.  MODE     ,I4/)
  510 FORMAT  (    // 10X, 36HPRESSURES DUE TO SYMMETRIC  GUST MODE,I4/)
  520 FORMAT  (    // 10X, 36HPRESSURES DUE TO ANTISYMM.  GUST MODE,I4/)
  530 FORMAT  (  //10X,40HZ-BODY FORCES DUE TO SYMMETRIC  GUST MODE,I4/)
  540 FORMAT  (  //10X,40HZ-BODY FORCES DUE TO ANTISYMM.  GUST MODE,I4/)
  550 FORMAT  (  //10X,40HY-BODY FORCES DUE TO SYMMETRIC  GUST MODE,I4/)
  560 FORMAT  (  //10X,40HY-BODY FORCES DUE TO ANTISYMM.  GUST MODE,I4/)
C
C
C         COPY ANTISYMMETRIC SOLUTIONS ONTO TAPES
C
```

```
      REWIND  NTEMP                                                     PIST0418
      IF   (NB .EQ. 0)  GO TO   590                                     PIST0419
      IF   (NASYM .EQ. 0)  GO TO   590                                  PIST0420
      NTAPE     =  MASTFZ                                               PIST0421
      DO   580  IZY = 1, 2                                              PIST0422
      DO   570  NM = 1, NASYM                                           PIST0423
      REAC    (NTEMP)  (DFQA(I),   I = 1, NSBETO)                       PIST0424
      WRITE   (NTAPE)  (DFQA(I),   I = 1, NSBETO)                       PIST0425
  570 CONTINUE                                                          PIST0426
C                                                                       PIST0427
      NTAPE     =  MASTFY                                               PIST0428
  580 CONTINUE                                                          PIST0429
  590 CONTINUE                                                          PIST0430
C                                                                       PIST0431
      NMTO      =  NASYM + NR                                           PIST0432
      DO   600  NM = 1, NMTO                                            PIST0433
      REAC    (NTEMP)  (DCPA(I),   I = 1, NBOX)                         PIST0434
      WRITE   (MASTCP) (DCPA(I),   I = 1, NBOX)                         PIST0435
  600 CONTINUE                                                          PIST0436
C                                                                       PIST0437
      IF   (NB .EQ. 0)  GO TO   630                                     PIST0438
      DO   610  NM = 1, NR                                              PIST0439
      READ    (NTEMP)  (DFQA(I),   I = 1, NSBETO)                       PIST0440
      WRITE   (MASTFZ) (DFQA(I),   I = 1, NSBETO)                       PIST0441
      READ    (NTEMP)  (DFQA(I),   I = 1, NSBETO)                       PIST0442
      WRITE   (MASTFY) (DFQA(I),   I = 1, NSBETO)                       PIST0443
  610 CONTINUE                                                          PIST0444
C                                                                       PIST0445
  630 CONTINUE                                                          PIST0446
C                                                                       PIST0447
      RETURN                                                            PIST0448
C                                                                       PIST0449
      END                                                               PIST0450
```

```
      SUBROUTINE  READD ( D, N, NTAPE)                                  READD  2
CREADD          SUBROUTINE TO READ DZ OR DY FROM TAPE                   READD  3
C               USE OF AN ARRAY TYPE OF INPUT TO INCREASE SPEED         READD  4
      COMPLEX   D(N)                                                    READD  5
      READ ( NTAPE ) IROW, IT, D                                        READD  6
      RETURN                                                            READD  7
      END                                                               READD  8
```

```
      SUBROUTINE RHSIDE( NTOT, NMODE, JM, IX, MK1, MK2, LENGTH,          RHSIDE 2
     1                   RHS   ,  WORK                     )             RHSIDE 3
      COMPLEX    WORK(LENGTH), RHS(NMODE)                                RHSIDE 4
C                                                                        RHSIDE 5
      DO 10   JX=1,NMODE                                                 RHSIDE 6
      JM    = JM+1                                                       RHSIDE 7
      JJ        =  (JM - 1) * NTOT + IX                                  RHSIDE 8
C                                                                        RHSIDE 9
      RHS(JX) =  WORK(JJ)                                                RHSIDE10
      IF  (IX.GE.MK1. AND .IX.LE.MK2)    RHS(JX) = (0.0,0.0)            RHSIDE11
   10 CONTINUE                                                           RHSIDE12
C                                                                        RHSIDE13
      RETURN                                                             RHSIDE14
      END                                                                RHSIDE15
```

```
      SUBROUTINE      ROWDYZ    (NFB    ,    NLB    ,    ROW    ,                 ROWDYZ  2
     1                NTZYS ,    D, DA  ,    DX     ,    DY     ,    DZ,   IRB,    ROWDYZ  3
     2     BETA, IDZDY, NTAPE, NTAP2, SGR, CGR, IPRNT                               ROWDYZ  4
     3 ,           NSBE, AO, YB, ZB, AR, XIS1, XIS2 )                               ROWDYZ  5
      DIMENSION                 NSBE(1), AO(1), YB(1), ZB(1), AR(1)                 ROWDYZ  6
      DIMENSION XIS1(1), XIS2(1)                                                    ROWDYZ  7
CROWDYZ         CALCULATE A ROW OF DZ OR  DY                                        ROWDYZ  8
C               SLENDER BODY                                                        ROWDYZ  9
C                                                                                   ROWDYZ10
C  *** *    *     *     *    *    *    *    *    *    *    *    *    *    *         ROWDYZ11
C  *        NFB         FIRST BODY OF THE DESIRED ORIENTATION - Z OR Y -  *         ROWDYZ12
C  *        NLB         LAST  BODY OF THE DESIRED ORIENTATION             *         ROWDYZ13
C  *        ROW         ROW OF DZ OR DY BEING CALCULATED                  *         ROWDYZ14
C  *        NTZYS       NO. COLUMNS TO BE CALCULATED                      *         ROWDYZ15
C  *        D           CALCULATED ROW                                    *         ROWDYZ16
C  *        DX          X - COORD. OF RECEIVING POINT                     *         ROWDYZ17
C  *        DY          Y - ...                                           *         ROWDYZ18
C  *        DZ          Z - ...                                           *         ROWDYZ19
C  *        BETA        SQRT ( 1 - M**2 )                                 *         ROWDYZ20
C  *        MACH        M - MACH NO.                                      *         ROWDYZ21
C  *        IDZDY       FLAG REQUIRED FOR FLLD                            *         ROWDYZ22
C  *** *    *     *     *    *    *    *    *    *    *    *    *    *    *         ROWDYZ23
      COMMON /AROCOM/ NTI, MODES                                                    AROCOM  2
     X , NP, MSTRIP, NSMAX, NCMAX, NTOTAL, NB, MSBE, MBE                            AROCOM  3
     Y , ND, NE, NBY, NBZ, NTO, NTP, NTY, NTZ                                       AROCOM  4
     1 , NTYS, NTZS, MAXGR, MAXSTR, NSBETO, NSTRIP, KR, XM, REFA, REFC              AROCOM  5
     2 , REFS, FMACH, LINES                                                         AROCOM  6
C                                                                                   ROWDYZ25
C                                                                                   ROWDYZ26
      DIMENSION  D(2, NTZYS),  DA(2, NTZYS)                                         ROWDYZ27
      REAL       KR                                                                 ROWDYZ28
      INTEGER    B1,  T1,  B,  T, ROW                                               ROWDYZ29
      DATA   NPOT  /6/                                                              ROWDYZ30
      DELTA     =   1                                                               ROWDYZ31
      DELT2     =  -1                                                               ROWDYZ32
C                                                                                   ROWDYZ33
      EPSLON   =       NE                                                           ROWDYZ34
      B1       =   0                                                                ROWDYZ35
      T1       =   0                                                                ROWDYZ36
                                                                                    ROWDYZ37
      DO     100    B =  NFB,  NLB                                                  ROWDYZ38
      B1       =   B1  +  1                                                         ROWDYZ39
C                                                                                   ROWDYZ40
      DAR      =   AR(B)                                                            ROWDYZ41
      NSBEB    =       NSBE ( B )                                                   ROWDYZ42
      IF ( IPRNT .NE.  0 )  WRITE (NPOT,130)  B, DY,YB(B), DZ, ZB(B)                ROWDYZ43
C                                                                                   ROWDYZ44
C                                                                                   ROWDYZ45
                                                                                    ROWDYZ46
C                   LOOP FOR EACH FLEMENT IN BODY -B-                               ROWDYZ47
      DO     80   T = 1, NSBEB                                                      ROWDYZ48
      T1       =    T1 +  1                                                         ROWDYZ49
```

```
      D( 1, T1)       =     0.0                                               RCWDYZ50
      D( 2, T1)       =     0.0                                               ROWDYZ51
      DA(1, T1)       =     0.0                                               ROWDYZ52
      DA(2, T1)       =     0.0                                               ROWDYZ53
      LHS        =  0                                                         ROWDYZ54
      XI1       =    XIS1( T1 )                                               ROWDYZ55
      XI2       =    XIS2( T1 )                                               ROWDYZ56
      AZRO      =      AO(T1)                                                 ROWDYZ57
      ETA        =     YB( B)                                                 ROWDYZ58
      ZETA       =     ZB( B )                                                ROWDYZ59
C       CHECK TO SEE IF CALCULATIONS ARE TO BE MADE                           ROWDYZ60
      IF  (   (  DY .EQ. ETA )  .AND.  ( DZ  .EQ.  ZETA )   )  GO TO 20       ROWDYZ61
C                                                                             ROWDYZ62
C                           ASSIGN RETURN ADDRESS FROM DZY                    ROWDYZ63
      ASSIGN    10 TO   JDZDY                                                 ROWDYZ64
      GO  TO    110                                                           ROWDYZ65
   10 D (1, T1)       =  DZYR                                                 ROWDYZ66
      D (2, T1)       =  DZYI                                                 ROWDYZ67
      DA(1, T1)       =  DZYR                                                 ROWDYZ68
      DA(2, T1)       =  DZYI                                                 ROWDYZ69
C                                                                             ROWDYZ70
C                           SKIP  IF NO SYMMETRY                              ROWDYZ71
   20 CONTINUE                                                                ROWDYZ72
      IF (  DELTA .EQ. 0.0 )  GO TO     60                                    ROWDYZ73
      ETA             =  -YB( B)                                              ROWDYZ74
C      IF SENDING BODY LIES ON Z AXIS.. SKIP SYMMETRY                         ROWDYZ75
      IF (  ETA  .EQ.  0.0  )  GO TO 60                                       ROWDYZ76
C       CHECK TO SEE IF CALCULATIONS ARE TO BE MADE                           ROWDYZ77
      IF  (   (  DY .EQ. ETA )  .AND.  ( DZ  .EQ.  ZETA )   )  GO TO 40       ROWDYZ78
      LHS        =  1                                                         ROWDYZ79
C                                  RETURN ADDRESS FROM DZY                    ROWDYZ80
      ASSIGN    30 TO   JDZDY                                                 ROWDYZ81
      GO TO     110                                                           ROWDYZ82
   30 D (1, T1)       =     D(1, T1)  +  DELTA * DZYR                         ROWDYZ83
      DA(1, T1)       =     DA(1,T1)  +  DELT2 * DZYR                         ROWDYZ84
      D (2, T1)       =     D(2, T1)  +  DELTA * DZYI                         ROWDYZ85
      DA(2, T1)       =     DA(2,T1)  +  DELT2 * DZYI                         ROWDYZ86
C                                                                             ROWDYZ87
   40 CONTINUE                                                                ROWDYZ88
      IF ( EPSLON .EQ. 0.0 )  GO TO     80                                    ROWDYZ89
C                     CALC. ONLY IF DELTA AND EPSLON  NOT EQUAL ZERO          ROWDYZ90
      ETA             =  -YB( B)                                              ROWDYZ91
      ZETA            =   -ZB(B )                                             ROWDYZ92
C     IF SENDING BODY LIES ON Z OR Y AXIS..  SKIP GRND. AND SYMMETRY          ROWDYZ93
      IF  (   ZETA  .EQ.  0.0  )            GO TO 60                          ROWDYZ94
C       CHECK TO SEE IF CALCULATIONS ARE TO BE MADE                           ROWDYZ95
      IF  (   (  DY .EQ. ETA )  .AND.  ( DZ  .EQ.  ZETA )   )  GO TO 60       ROWDYZ96
      LHS        =  1                                                         ROWDYZ97
C                              ASSIGN RETURN ADDRESS FROM DZY                 ROWDYZ98
      ASSIGN    50 TO   JDZDY                                                 ROWDYZ99
      GO  TO    110                                                           ROWDY100
   50 D (1,   T1)     =  D( 1, T1 ) + EPSLON* DELTA * DZYR                    ROWDY101
```

```
      DA(1,   T1)      =   DA(1,  T1 ) + EPSLON* DELT2 * DZYR                   ROWDY102
      D (2,   T1)      =   D( 2,  T1 ) + EPSLON* DELTA * DZYI                   ROWDY103
      DA(2,   T1)      =   DA(2,  T1 ) + EPSLON* DELT2 * DZYI                   ROWDY104
C                                                                               ROWDY105
C                               SKIP IF NO GROUND EFFECTS                       ROWDY106
   60 IF ( EPSLON   .EQ. 0.0 ) GO    TO    80                                   ROWDY107
      ETA      =      YB(B)                                                     ROWDY108
      ZETA     =     -ZB(B)                                                     ROWDY109
C     IF SENDING BODY LIES ON THE Y AXIS..  SKIP GROUND EFFECTS                 ROWDY110
      IF (  ZETA   .EQ.   0.0   )  GO TO 80                                     ROWDY111
C       CHECK TO SEE IF CALCULATIONS ARE TO BE MADE                             ROWDY112
      IF  (   (  DY .EQ. ETA )   .AND.   ( DZ   .EQ.   ZETA )   )  GO TO 80     ROWDY113
C                                ASSIGN RETURN ADDRESS FROM DZY                 ROWDY114
C                                                                               ROWDY115
      ASSIGN    70 TO    JDZDY                                                  ROWDY116
      GO  TO    110                                                             ROWDY117
   70 D (1, T1)        =     D(1, T1)   +   EPSLON * DZYR                       ROWDY118
      DA(1, T1)        =     DA(1,T1)   +   EPSLON * DZYR                       ROWDY119
      D (2, T1)        =     D(2, T1)   +   EPSLON * DZYI                       ROWDY120
      DA(2, T1)        =     DA(2,T1)   +   EPSLON * DZYI                       ROWDY121
   80 CONTINUE                                                                  ROWDY122
C                        END OF LOOP FOR ELEMENT                                ROWDY123
   90 CONTINUE                                                                  ROWDY124
C                200 IS END OF LOOP ON SLENDER BODY                             ROWDY125
C                                                                               ROWDY126
  100 CONTINUE                                                                  ROWDY127
C                                                                               ROWDY128
C              WRITE ROW ON TAPE,  ROW NUMBER, NO. ELEMENTS, DATA               ROWDY129
      WRITE (NTAPE )  ROW, T1, D                                                ROWDY130
      WRITE  (NTAP2 )  ROW, T1, DA                                              ROWDY131
      IF ( IPRNT .NE. 0 )  WRITE (NPOT,120)  ROW, T1, D                         ROWDY132
      RETURN                                                                    ROWDY133
C                                                                               ROWDY134
C                      CALL SEQUENCE TO DZY                                     ROWDY135
C                                                                               ROWDY136
  110 CALL  DZY (     DX     ,    DY     ,    DZ     ,    SGR    ,    CGR    ,  ROWDY137
     1                XI1    ;    XI2    ;    ETA    ;    ZETA   ;    DAR    ,  ROWDY138
     2     AZRC   ,   KR     ,    REFC   ,    BETA   ,                          ROWDY139
     3      FMACH,     IDZDY,      DZYR,      DZYI   ;    LHS    ,             ROWDY140
     4     IPRNT          )                                                     ROWDY141
      LHS          =   0                                                        ROWDY142
C                                                                               ROWDY143
C                                                                               ROWDY144
      GO TO     JDZDY , (10,30,50,70)                                           ROWDY145
C                                                                               ROWDY146
  120 FORMAT (18H ROWDYZ--- ROW NO., I5, 1H,, I10,10H ELEMENTS. /               ROWDY147
     1      ( 6E12.4)   )                                                       ROWDY148
  130 FORMAT (12H ROWDYZ  B =, I10, 4E20.8 )                                    ROWDY149
  140 FORMAT  (1H , Z10   )                                                     ROWDY150
      END                                                                       ROWDY151
```

```
      SUBROUTINE RWREC(IFLAG, NTAPE,   A   , NCWORD, NUMBR, NEWTAP )
C    ***   RWREC
C
C    ***   READS AND/OR WRITES AN UNFORMATTED RECORD   A   ON TAPE
C          ACCORDING TO EITHER OF THE FOLLOEING THREF OPTIONS
C          IFLAG = 0   WRITE ARRAY  A   ON  TAPE  NTAPE
C          IFLAG = 1   READ  ARRAY  A  FROM TAPE  NTAPE
C          IFLAG = 2  COPY   NUMBR  OF ARRAYS   A  (EACH  NCWORD  LONG)
C    ***              FROM TAPE  NTAPE  ONTO TAPE  NEWTAP
C
      COMPLEX     A(NCWORD)
      IFLP1    =   IFLAG + 1
      GO TO  (10, 20, 30),   IFLP1
   10 WRITE  (NTAPE)  A
      RFTURN
   20 READ   (NTAPE)  A
      RETURN
   30 CONTINUE
      DO  40  N = 1, NUMBR
      READ   (NTAPE)  A
      WRITE  (NEWTAP) A
   40 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SB (NM, NSYM, NASYM, NTP6, NTPH, NTPDH, IPRINT         SB       2
     1 , WORK, DW, CPZ A, UZYA                                          SB       3
     1 , NBEA,        NSBE, NC, NS, AO, YB, ZB, AR, AOP, XIS1, XIS2, X  SB       4
     2 , CG, SG, YP, ZP, CR, NG )                                       SB       5
C                                                                       SB       6
      COMPLEX WORK(1), DW(1), CPZYA(1), UZYA(1)                         SB       7
      DIMENSION NBEA(2 ,1),            NSBE(1), NC(1), NS(1), AO(1)     SB       8
     1 , YB(1), ZB(1), AR(1), AOP(1), XIS1(1), XIS2(1), X(1), CG(1)     SB       9
     2 , SG(1), YP(1), ZP(1), CK(3,1)                                   SB      10
C                                                                       SB      11
C   ***   CALCULATES THE NORMALWASH ON LIFTING SURFACE BOXES AND        SB      12
C         INTERFERENCE BODY ELEMENTS DUE  TO SLENDER BODY ELEMENTS      SB      13
C *                                                                  *  SB      14
C *       ********* ***********************************************  SB      15
C *                                                                  *  SB      16
C *       NM         TOTAL NUMBER OF MODES FOR BODIES                   SB      17
C *       NSYM       NUMBER OF SYMMETRIC MODES                          SB      18
C *       NASYM      NUMBER OF ANTISYMMETRIC MODES                      SB      19
C *       WORK       COMPLEX WORKING ARRAY                              SB      20
C *                                                                  *  SB      21
C *       ********* ***********************************************  SB      22
C *                                                                  *  SB      23
      COMMON /AROCOM/ NTI, MODES                                        AROCOM   2
     X , NP, MSTRIP, NSMAX, NCMAX, NTOTAL, NB, MSBE, MBE                AROCOM   3
     Y , ND, NE, NBY, NBZ, NTO, NTP, NTY, NTZ                           AROCOM   4
     1 , NTYS, NTZS, MAXGR, MAXSTR, NSBETO, NSTRIP, KR, XM, REFA, REFC  AROCOM   5
     2 , REFS, FMACH, LINES                                             AROCOM   6
C                                                                       SB      25
      REAL      KR                                                      SB      26
C                                                                       SB      27
      NUMBR    =  0                                                     SB      28
      NR       =  0                                                     SB      29
      BETA     =  SQRT(1.0 - FMACH**2)                                  SB      30
      N        =  NTP + NTZ + NTY                                       SB      31
      NFZB     =  1                                                     SB      32
      NLZB     =  NBZ                                                   SB      33
      NFYB     =  NB + 1 - NBY                                          SB      34
      NLYB     =  NB                                                    SB      35
      NDZ      =  1                                                     SB      36
      NDY      =  2                                                     SB      37
      NDZ2     =  10                                                    SB      38
      NDY2     =  11                                                    SB      39
      NW1      =  3                                                     SB      40
      NW2      =  9                                                     SB      41
      NW3      =  4                                                     SB      42
      NW4      =  12                                                    SB      43
      NW5      =  13                                                    SB      44
      NW6      =  14                                                    SB      45
      REWIND   NW1                                                      SB      46
      REWIND   NW2                                                      SB      47
      REWIND  NW3                                                       SB      48
      REWIND  NW4                                                       SB      49
```

```
      REWIND  NW5
      REWIND  NW6
C                  CALCULATE DZ AND DY MATRICES
C
      REWIND  NDZ
      REWIND  NDZ2
C                        CALCULATE DZ
      IDZDY    =     0
      LOC        =  NTZS + 1
      CALL  DZYMAT(WORK(1), WORK(LOC), NFZB, NLZB, NTZS, IDZDY,
     1            NDZ, NDZ2,     X    , BETA
     2 , NUMBR
     2 , NBEA,      NSBE,NC,NS,AO,YB,ZB,AR,XIS1,XIS2,CG,SG,YP,ZP )
C
C                        CALCULATE DY
C
      REWIND  NDY
      REWIND  NDY2
      IDZDY    =     1
      LOC        =  NTYS + 1
      CALL  DZYMAT(WORK(1), WORK(LOC), NFYB, NLYB, NTYS, IDZDY,
     1            NDY, NDY2,     X    , BETA
     2 , NUMBR
     2 , NBEA,      NSBE,NC,NS,AO,YB,ZB,AR,XIS1,XIS2,CG,SG,YP,ZP )
C
C
   10 CONTINUE
C
      NTZY     =      NTZS + NTYS
      NFW      =      1
      NFDZ     =      NFW  +  N
      NFDY     =      NFDZ +  N * NTZS
      NFUZ     =      NFDY +  N * NTYS
      NFUY     =      NFUZ +  NTZS
      NFCPZ    =      NFUY +  NTYS
      NFCPY    =      NFCPZ+  NTZS
      NFE      =      NFCPY+  NTYS
      IEND     =      NFE - 1
      NFH = NFE + NTZS
      NFDH = NFH + N
C
      IGO      =     1
      ASSIGN  70 TO  IZ
      ASSIGN  130 TO  IY
   20 CONTINUE
      J          =  0
      REWIND  NDZ
      REWIND  NDY
C                  READ DZ AND DY FROM TAPE
C
      N1       =    NFDZ
      DO     30    I  =  1, N
```

```
      CALL READD ( WORK(N1), NTZS , NDZ )
      N1      =     N1 +  NTZS
   30 CONTINUE
      N1      =     NFDY
      DO    40    I  =  1, N
      CALL READD ( WORK(N1), NTYS , NDY )
      N1      =     N1 +  NTYS
   40 CONTINUE
   42 CONTINUE
C                  NOW CALCULATE UZ AND UY A COLUMN AT A TIME AND
C                  MULTIPLY
C
      DO  50  IWK = NFUZ , IEND
      WORK(IWK) = 0.0
   50 CONTINUE
C
C
      NFEND     =          MAXO(NTZS, NTYS)
      CALL          ZERUUT(WORK(NFE ), NFEND, 1 ,  0 )
   60 CONTINUE
      READ    (NTPH)  NODE
      READ    (NTPDH)
      IF   (NODE .EQ. -1)  GO TO  150
      J          =  J + 1
      GO TO   IZ    , (70,90)
C
C                           CALCULATE UZ COLUMN
   70 CONTINUE
      CALL  MUZYC ( J      ,     2     ,    NTZS   ,    NFZB   ,   NLZB   ,
     1                NSBE   ,    KR     ,               REFC   ,   AO     ,
     2                AOP    ,    XIS1   ,    XIS2   ,    AR     ,   NTPH   ,
     3 NTPDH, WORK(NFUZ), WORK(NFCPZ),
     4   UZYA, CPZYA, NR, FMACH, YB, ZB, CR, NG, WORK(NFH), WORK(NFDH) )
      IEND  =   NFE - 1
C
      CALL          ZEROUT(WORK(NFZB), N     ,  1 ,  0 )
      CALL          DUMULT(1,  N,  NTZS, 0 , WORK( 1 ),
     1                     WORK(NFDZ) , WORK(NFUZ) , 0  ,   0  )
C
      WRITE   (NW3 )  NODE
      CALL          RWREC(  0,  NW3,  WORK       ,  N          , 1,   0)
      NFCEND    =  NFCPZ + NTZS - 1
      JJEND     =  NFE + NTYS - 1
      NFUZYL    =  NFUY - 1
      WRITE   (NW5)   ((WORK(II), II = NFCPZ, NFCEND),
     1                  (WORK(JJ),  JJ = NFE, JJEND))
C
C                           CALCULATIONS FOR UY COLUMN
C
   90 GO TO IY, (100,130)
C
  100 CONTINUE
```

```
      CALL  MUZYC ( J      ,    3     ,   NTYS   ,   NFYB   ,   NLYB   ,    SB
     1              NSBE   ,    KR    ,          ,   REFC   ,   AO     ,    SB
     2              AOP    ,    XIS1  ,   XIS2   ,   AR     ,   NTPH   ,    SB
     3   NTPDH, WORK(NFUY), WORK(NFCPY),                                    SB
     4   UZYA, CPZYA, NR, FMACH, YB, ZB, CR, NG, WORK(NFH), WORK(NFDH) )SB
  110 CONTINUE                                                              SB
      CALL         ZEROUT(WORK(NFZB), N    ,  1 ,  0 )                      SB
      CALL         DUMULT(1,  N,  NTYS, 0 , WORK( 1 ),                      SB
     1                    WORK(NFDY) , WORK(NFUY) , 0  ,  0  )              SB
      WRITE  (NW4 )  NODE                                                   SB
      CALL         RWREC(  0,  NW4,  WORK      ,  N        , 1,   0)        SB
      NFCEND    =  NFCPY + NTYS - 1                                         SB
      JJEND     =  NFE + NTZS - 1                                           SB
      NFUZYL    =  NFUY + NTYS - 1                                          SB
      WRITE  (NW6)  ((WORK(JJ),  JJ = NFE, JJEND),                          SB
     1               (WORK(II), II = NFCPY, NFCEND))                        SB
  130 CONTINUE                                                              SB
C                                                                           SB
  140 CONTINUE                                                              SB
      GO TO  60                                                             SB
  150 CONTINUE                                                              SB
      GO TO  (160,180,170,200),   IGO                                       SB
  160 CONTINUE                                                              SB
      NDZOLD    =  NDZ                                                      SB
      NDYOLD    =  NDY                                                      SB
      IF  (NASYM .EQ. 0)  GO TO  180                                        SB
      WRITE  (NW3)  NODE                                                    SB
      GO TO  172                                                            SB
  170 CONTINUE                                                              SB
      IF  (NASYM .EQ. 0)  GO TO  200                                        SB
      WRITE  (NW4)  NODE                                                    SB
  172 NDZ       =  NDZ2                                                     SB
      NDY       =  NDY2                                                     SB
      IGO       =  IGO + 1                                                  SB
      GO TO  20                                                             SB
  180 WRITE  (NW3)  NODE                                                    SB
      NDZ       =  NDZOLD                                                   SB
      NDY       =  NDYOLD                                                   SB
      IGO       =  3                                                        SB
      READ   (NTPH)                                                         SB
      READ   (NTPDH)                                                        SB
      ASSIGN  90 TO  IZ                                                     SB
      ASSIGN  100 TO  IY                                                    SB
      IF  (NASYM .EQ. 0)  GO TO  42                                         SB
      GO TO  20                                                             SB
  200 WRITE  (NW4)  NODE                                                    SB
      NDZ       =  NDZOLD                                                   SB
      NDY       =  NDYOLD                                                   SB
C                                                                           SB
      NM        =  NSYM + NASYM                                             SB
      REWIND  NW3                                                           SB
      REWIND  NW4                                                           SB
```

```
      REWIND  NW5
      REWIND  NW6
      DO  300   MD = 1, NM
      READ    (NW3)   JCOL
      READ    (NW3)   ( DW (L),   L = 1,   N)
      READ    (NW4)
      READ    (NW4)   (WORK(L),   L = 1,   N)
      DO  280  L = 1,  N
      DW(L)     =   DW(L)   + WORK(L)
  280 CONTINUE
C
      READ (NW5) (CPZYA(L), L=1, NTZY)
      READ   (NW6)   (WORK(L),   L = 1, NTZY)
      DO  290  L = 1, NTZY
      CPZYA(L) = CPZYA(L) + WORK(L)
  290 CONTINUE
      WRITE   (NW1)   JCOL
      WRITE   (NW1)   ( DW (L),   L = 1,   N)
      WRITE (NW2) (CPZYA(L), L=1, NTZY)
      IF   (MD .NE. NSYM)  GO TO  270
      READ    (NW3)   KODE
      READ    (NW4)
      WRITE   (NW1)   KODE
  270 CONTINUE
C
      IF   (IPRINT .LT. 2)  GO TO  300
      WRITE (NTP6,310) MD, NTZY, (CPZYA(L), L=1,NTZY)
C
  300 CONTINUE
      IF   (NASYM .NE. 0)  WRITE  (NW1)  KODE
C
      REWIND  NW3
      REWIND  NW5
      REWIND  NW6
      NDZ         =   NDZOLD
      NDY         =   NDYOLD
      IGO         =   1
      L2          =   NTZS + 1
C
  450 CONTINUE
      REWIND NDZ
      REWIND NDY
      N1          =   NFDZ
      DO  460   I = 1, N
      CALL          READD(WORK(N1),  NTZS, NDZ)
      N1          =   N1 + NTZS
  460 CONTINUE
      N1          =   NFDY
      DO  480   I = 1, N
      CALL          READD(WORK(N1), NTYS, NDY)
      N1          =   N1 + NTYS
  480 CONTINUE
```

```
C
      DO  560  NR = 1, NG
      J          =  J + 1
      CALL         ZEROUT(WORK(NFZB), N, 1, 0 )
      GO TO  (500, 520),   IGO
  500 CONTINUE
      CALL         MUZYC(  J, 2, NTZS,          NFZB, NLZB, NSBE, KR , REFC,
     1             AO ,  AOP,      XIS1, XIS2, AR   , NTPH, NTPDH          ,
     2             WORK(NFUZ),  WORK(NFCPZ), UZYA( 1 ),  CPZYA( 1 )      ,
     3       NR, FMACH, YB, ZB, CR, NG, WORK(NFH), WORK(NFDH) )
C
      CALL         MUZYC(  J, 3, NTYS,          NFYB, NLYB, NSBE, KR , REFC,
     1             AO , AOP,       XIS1, XIS2, AR   , NTPH, NTPDH          ,
     2             WORK(NFUY) , WORK(NFCPY), UZYA( L2),  CPZYA( L2)      ,
     3       NR, FMACH, YB, ZB, CR, NG, WORK(NFH), WORK(NFDH) )
C
      CALL         ZEROUT(WORK(NFZB), N, 1, 0 )
C
      CALL         DUMULT( 1, N, NTZS, NTYS, WORK( 1 ),
     1             WORK(NFDZ), WORK(NFUZ),    WORK(NFDY), WORK(NFUY)        )
C
      WRITE  (NW1)  (WORK(LW),  LW = 1, N )
      WRITE  (NW3)  (WORK(LC),  LC = NFCPZ, NFCEND)
      WRITE  (NW5)  (UZYA (L),  L  = 1, NTZY)
      WRITE  (NW6)  (CPZYA(L),  L  = 1, NTZY)
      GO TO  560
  520 CONTINUE
C
      READ   (NW5)  (UZYA (L),  L = 1, NTZY)
C
      CALL         DUMULT(1, N, NTZS, NTYS, WORK(1),
     1             WORK(NFDZ), UZYA( 1 ),   WORK(NFDY), UZYA( L2)         )
C
      WRITE  (NW1)  (WORK(LW),  LW = 1, N)
  560 CONTINUE
      IF  (IGO .EQ. 2 .OR. NASYM .EQ. 0)  GO TO  700
      REWIND  NW5
      NDZ       =  NDZ2
      NDY = NDY2
      IGO       =  2
      GO TO  450
C
  700 CONTINUE
C
      REWIND  NW2
      REWIND  NW3
      REWIND  NW5
      REWIND  NW6
      CALL         RWREC(  2  ,  NW2, CPZYA, NTZY, NM   , NW5 )
      REWIND  NW2
      REWIND  NW5
      CALL         RWREC(  2  ,  NW5, CPZYA, NTZY, NSYM , NW2 )
```

```
      CALL RWREC(2,NW3,CPZYA,NTZY,NG,NW2)
      IF  (NASYM .EQ. 0)  GO TO  710
      CALL        RWREC(  2  ,  NW5, CPZYA, NTZY, NASYM, NW2 )
      CALL RWREC(2,NW6,CPZYA,NTZY,NG,NW2)
  710 CONTINUE
      REWIND  NW1
      REWIND  NW2
  310 FORMAT  ( // 9H   MODE =, I5 / 16H   NTZS + NTYS =, I5 /
     1                30H   CP-Z*DELTA-A, CP-Y*DELTA-A  // (6E14.6 ) )
C
      RETURN
      END
```

```
      SUBROUTINE SDLM (ELXI,ELYI,ELZI,PHIN,PHIZ,PHIY, NODES              SDLM   2
     1 , NSYM, NASYM, MFIX1, MFIX2, IFNEWH, MDOF, PHI, MDOFP, MDOFB      SDLM   3
     2 , NAS, NASB, NBARAY, NCARAY, NSARAY, ISSTR, NSSTR                 SDLM   4
     3 , IFLA, NBEA, NT12, NSBEA                                         SDLM   5
     4 , YB, ZB, ARB, AVR, XLE, XTE, RIA, TH1A, AO, AOP, XIS1, XIS2      SDLM   6
     5 , CG, CS, EE, SG, YS, ZS, XIJ, YIN, ZIN, COORD                    SDLM   7
     6 , X, XIC, DELX, XLAM  H, DHDX, DELA                               SDLM   8
     7 , DT, DTA, RHS, XC )                                              SDLM   9
C    ***                                                                 SDLM  10
C     SUBSONIC UNSTEADY AERODYNAMICS FOR GENERAL CONFIGURATIONS          SDLM  11
C     METHOD CALCULATES LIFTING PRESSURE, SPAN LOADS AND MOMENTS,        SDLM  12
C     GENERALIZED FORCES, AND OTHER AERODYNAMIC CHARACTERISTICS FOR      SDLM  13
C     GENERAL LIFTING SURFACE - BODY COMBINATIONS.  STEADY AND UNSTEADY SDLM  14
C     FLOWS ARE CONSIDERED. BODIES MAY HAVE ELLIPTIC CROSS SECTIONS.     SDLM  15
C     LIFTING SURFACES MAY HAVE ARBITRARY PLANFORM AND DIHEDRAL.         SDLM  16
C     CONTROL SURFACES ARE HANDLED.  PROGRAM IS LARGE ENOUGH TO HANDLE  SDLM  17
C     ENTIRE AIRCRAFT.                                                   SDLM  18
C                                                                        SDLM  19
C    ***                                                                 SDLM  20
      COMMON NAA,WORK(1)                                                 SDLM  21
      COMMON /AROCOM/ NTI, MODES                                         AROCOM 2
     X , NP, MSTRIP, NSMAX, NCMAX, NTOTAL, NB, MSBE, MBE                 AROCOM 3
     Y , ND, NE, NBY, NBZ, NTO, NTP, NTY, NTZ                            AROCOM 4
     1 , NTYS, NTZS, MAXGR, MAXSTR, NSBETO, NSTRIP, KR, XM, REFA, REFC   AROCOM 5
     2 , REFS, FMACH, LINES                                              AROCOM 6
      REAL      KR                                                       SDLM  23
      DIMENSION ELXI(1), ELYI(1), ELZI(1)                                SDLM  24
      DIMENSION PHIN(NODES,1), PHIZ(NODES,1), PHIY(NODES,1)              SDLM  25
      DIMENSION  PHI(1)                                                  SDLM  26
      DIMENSION NAS(1), NASB(1), NBARAY(1), NCARAY(1), NSARAY(1)         SDLM  27
      DIMENSION ISSTR(1), NSSTR(1)                                       SDLM  28
      DIMENSION IFLA(2,1), NBEA(2,1), NT12(2,1), NSBEA(1)                SDLM  29
      DIMENSION YB(1), ZB(1), ARB(1), AVR(1), XLE(1), XTE(1)             SDLM  30
      DIMENSION RIA(1), TH1A(1), TH2A(1)                                 SDLM  31
      DIMENSION AO(1), AOP(1), XIS1(1), XIS2(1)                          SDLM  32
      DIMENSION CG(1), CS(1), EE(1), SG(1), YS(1), ZS(1)                 SDLM  33
      DIMENSION XIJ(1), YIN(1), ZIN(1), COORD(1)                         SDLM  34
      DIMENSION X(1), XIC(1), DELX(1), XLAM(1)                           SDLM  35
      DIMENSION H(1), DHDX(1), DELA(1)                                   SDLM  36
      COMPLEX  DT(1),DTA(1),RHS(1)                                       SDLM  37
      DIMENSION FREQ(100)                                                SDLM  38
      DIMENSION IHD(50),RHD(50)                                          SDLM  39
      EQUIVALENCE (IHD(1),RHD(1))                                        SDLM  40
      DIMENSION NMODES(6), CR(3,20), NTAPE(20)                           SDLM  41
      DIMENSION XC(1)                                                    SDLM  42
      DATA      NTAPE/1, 2, 3, 4, 5, 6, 7, 8, 9,10,11,12,13,14,6*0    /SDLM  43
      DATA      IWRITE, IREAD, ICOPY  / 0, 1, 2 /                        SDLM  44
      DATA NSBMIN, NUMBR, IZERO / 1, 0, 0 / ,                            SDLM  45
     1          NTPH  , NTPDH , NTPH4 , MASTDT, MASTCP, MASTFZ, MASTFY,SDLM  46
     2          NTAERO, NTGF   / 12, 13, 14, 15, 16, 17, 18, 19, 20 /    SDLM  47
      DATA      MASTH, MASTDH, MASTH4  / 21, 22, 23 /                    SDLM  48
      DATA      NDMINB, NDMINP, NMODES / 1, 1, 6*0  /                    SDLM  49
```

```
      DATA CR/  0.8660254, 0.0      , 0.50                         SDLM  50
     1  0.50     , 0.0      , 0.8660254,                             SDLM  51
     1  0.0      , 0.0      , 1.0      ,                             SDLM  52
     1 -0.7071068, 0.0      , 0.7071068,                             SDLM  53
     1 -0.7071068, 0.0      ,-0.7071068,                             SDLM  54
     1  0.0      , 0.0      ,-1.0      ,                             SDLM  55
     1  0.50     , 0.0      ,-0.8660254,                             SDLM  56
     1  0.8660254, 0.0      ,-0.50     ,                             SDLM  57
     1  0.7071068,-0.7071068,-0.0      ,                             SDLM  58
     1  0.0      ,-1.0      , 0.0      ,                             SDLM  59
     1 -0.7071068,-0.7071068, 0.0      ,                             SDLM  60
     1  0.0      ,-0.7071068, 0.7071068,                             SDLM  61
     1  0.0      ,-0.7071068,-0.7071068,                             SDLM  62
     1 0.0,0.0,0.0,                                                  SDLM  63
     1 0.0,0.0,0.0,                                                  SDLM  64
     1 0.0,0.0,0.0,                                                  SDLM  65
     1 0.0,0.0,0.0,                                                  SDLM  66
     1 0.0,0.0,0.0,                                                  SDLM  67
     1 0.0,0.0,0.0,                                                  SDLM  68
     1 0.0,0.0,0.0/                                                  SDLM  69
   10 FORMAT (6I12)                                                  SDLM  70
   20 FORMAT (6F12.0)                                                SDLM  71
   30 FORMAT (I12,3F12.0)                                            SDLM  72
   80 FORMAT  ( 1H1 /// 10X, 22H***   SEGMENT  5   ***  ///  5X,     SDLM  73
     1         65HSOLVE THE SYSTEM OF LINEAR EQUATIONS,   DT*DCP=WT,   FSDLM  74
     20R   -DCP-    //   )                                           SDLM  75
   90 FORMAT  (1H0,30X,36H**  ARRAY OF REDUCED FREQUENCIES   **  )   SDLM  76
  110 FORMAT  ( / 5X, 21HREFERENCE CHORD      = , F15.5              SDLM  77
     1          / 5X, 21HREFERENCE SEMI-SPAN  = , F15.5              SDLM  78
     2          / 5X, 21HREFERENCE AREA       = , F15.5              SDLM  79
     3          / 5X, 21HMACH NUMBER          = , F15.5              SDLM  80
     4          / 5X, 21HBETA                 = , F15.5              SDLM  81
     5          / 5X, 21HMOMENT AXIS          = , F15.5              SDLM  82
     6          / 5X, 21HIMAGE RADIUS         = , F15.5, 12H*BODY RADIUS SDLM  83
     7         // 5X, 21HNUMBER OF PANELS     = , I5                 SDLM  84
     8          / 5X, 21HNUMBER OF BODIES     = , I5                 SDLM  85
     9         // 5X, 21HNUMBER OF UNKNOWNS   = , I5  //  )          SDLM  86
  120 FCRMAT  (1H1 /// 23H      GUST OPTION FLAG =, I3 // 15X,       SDLM  87
     1         12HINPUT ARRAYS / 50H     R     COS(A-R)     COS(B-R) SDLM  88
     2 COS(G-R)                   / )                                SDLM  89
  130 FORMAT  ( I6, 6F13.5          )                                SDLM  90
  150 FORMAT  ( / 20X, 6F12.6   )                                    SDLM  91
C                                                                    SDLM  92
      LINES    =  45                                                 SDLM  93
      ND       =  0                                                  SDLM  94
      NDZ    = NTAPE(1)                                              SDLM  95
      NDY    = NTAPE(2)                                              SDLM  96
      ITAPE  = NTAPE(8)                                              SDLM  97
      NPTAP  = NTAPE(3)                                              SDLM  98
      NIN    = NTAPE(5)                                              SDLM  99
      NOUT   = NTAPE(6)                                              SDLM 100
      NPSTAP = NTAPE(9)                                              SDLM 101
```

```
      NM      = NTAPE(1)
      NO      = NTAPE(2)
      NDW     = NTAPE(3)
      NWT     = NTAPE(4)
      NWTA =  15
      NI      = NTAPE(10)
      NBFM    = NTAPE(11)
      NBFMA   = NTAPE(12)
      NEWBFZ  = NTAPE(13)
      NEWBFY  = NTAPE(14)
      NW      = NPTAP
      NE      = 0
C
      IF (IFNEWH.NE.0) GO TO 200
C
      CALL HEADNG
      WRITE (NOUT,50)
   50 FORMAT ( 1H0,4HNODE,8X7HELXI(I),8X7HELYI(I),8X7HELZI(I)/)
      DO 170 I=1,NODES
      READ (NTI,20) ELXI(I), ELYI(I), ELZI(I)
  170 WRITE (NOUT,40) I, ELXI(I), ELYI(I), ELZI(I)
   40 FORMAT ( 1X, I4,3E15.6)
C
      DO 185 J=1,MODES
C
      CALL HEADNG
      WRITE (NOUT,60) J,J,J
   60 FORMAT ( 1H0,4HNODE, 5X7HPHIN(I,I2,1H),5X7HPHIZ(I,I2,1H)
     1 , 5X7HPHIY(I,I2,1H)/)
      DO 180 I=1,NODES
      READ (NTI,20) PHIN(I,J), PHIZ(I,J), PHIY(I,J)
  180 WRITE (NOUT,40) I, PHIN(I,J), PHIZ(I,J), PHIY(I,J)
  185 CONTINUE
C
  200 CONTINUE
C
      READ (NTI,10) IPR1, IPR2, IPR3, NGUST
      READ (NTI,10) NKD, NKP, MK1, MK2
      NK    = NKD+NKP
      JRUN = 0
      IOPT = 0
      READ (NTI,20) FMACH, REFA, REFS, REFC, XM   , SCALER
      IF (SCALER.EQ.0.0) SCALER=2.0
      READ (NTI,20) (FREQ(I),I=1,NK)
C
      CALL HEADNG
      WRITE (NOUT,90)
      WRITE  (NOUT,150)  (FREQ(I),  I = 1, NK)
C
      BETA2  = 1.0 - FMACH**2
      IF  (BETA2 .LE. 0.0)  BETA2 = ABS(BETA2)
      BETA   = SQRT( BETA2 )
```

```
      WRITE  (NOUT,110)  REFC   , REFS   , REFA   , FMACH , BETA   ,      SDLM 154
     1     XM, SCALER,  NP,  NB,  NTOTAL                                SDLM 155
C                                                                       SDLM 156
      NG = 13                                                           SDLM 157
      IF (NGUST.NE.0) NG = NGUST                                        SDLM 158
      IF (MFIX2.GT.NSYM) MFIX2 = MFIX2+NG                               SDLM 159
      IF (MFIX1.GT.NSYM) MFIX1 = MFIX1+NG                               SDLM 160
      WRITE(NOUT,91)IPR1,IPR2,IPR3,NG                                   SDLM 161
   91 FORMAT(1H0,20X, IPR1 =   ,I2/1H ,                                 SDLM 162
     1 20X, IPR2 =   ,I2/1H ,                                           SDLM 163
     1 20X, IPR3 =   ,I2/1H ,                                           SDLM 164
     1 20X, NO. GUST ORIENTATIONS = ,I3)                                SDLM 165
C                                                                       SDLM 166
      IHD(2) = NK                                                       SDLM 167
      IHD(3) = NSYM                                                     SDLM 168
      IHD(4) = NASYM                                                    SDLM 169
      IHD(5) = NG                                                       SDLM 170
      IHD(6) = NB                                                       SDLM 171
      IHD(11) = NP                                                      SDLM 172
      RHD(21) = FMACH                                                   SDLM 173
      RHD(22) = REFA                                                    SDLM 174
      RHD(23) = REFS                                                    SDLM 175
      RHD(24) = REFC                                                    SDLM 176
      RHD(25) = XM                                                      SDLM 177
C                                                                       SDLM 178
      L1  = NAA                                                         SDLM 179
      L2 = L1+ NCMAX                                                    SDLM 180
      L3 = L2+ NSMAX                                                    SDLM 181
      L4 = L3+ MBE                                                      SDLM 182
      L5 = L4+ MBE                                                      SDLM 183
      L6 = L5+ MSBE                                                     SDLM 184
      L7 = L6+ MSBE                                                     SDLM 185
      L8 = L7+ NP                                                       SDLM 186
      L9 = L8+ MSTRIP+NB                                                SDLM 187
      L10 =  L9+ MSTRIP+NB                                              SDLM 188
      L11 = L10+ MSTRIP+NB                                              SDLM 189
      L12 = L11+ NTOTAL                                                 SDLM 190
      L13 = L12+ NTOTAL                                                 SDLM 191
      L14 = L13+ NTOTAL                                                 SDLM 192
      L15 = L14+ NTOTAL                                                 SDLM 193
      L16 = L15+ NTOTAL                                                 SDLM 194
      L17 = L16 + NTOTAL                                                SDLM 195
      L18 = L17 + NTOTAL                                                SDLM 196
      L19 = L18 + NTOTAL                                                SDLM 197
      L20 = L19 + MSBE                                                  SDLM 198
      L21 = L20 + MSBE                                                  SDLM 199
C                                                                       SDLM 200
      CALL       GEOM (DELA, IFLA, NBEA, NT12, NAS, NASB, ISSTR, NSSTR  SDLM 201
     1 , NSBEA, NBARAY, NCARAY, NSARAY, AO, YB, ZB, ARB, AVR, AOP, RIA  SDLM 202
     2 , XLE, XTE, THIA, XIS1, XIS2, X, CG, CS, EE, SG, YS, ZS, XIC, XIJSDLM 203
     3 , YIN, ZIN, DELX, XLAM, COORD, NOUT                              SDLM 204
     4 , WORK(L1), WORK(L2), WORK(L3), WORK(L4), WORK(L5), WORK(L6)     SDLM 205
```

```
5 , WORK(L7),WORK(L8),WORK(L9),WORK(L10)
6 , WORK(L11),WORK(L12),WORK(L13),WORK(L14),WORK(L15),WORK(L16)
* , XC
7 , WORK(L17),WORK(L18),WORK(L19),WORK(L20)
8 , IHD,WORK(L21),WORK(L21) )
C
C     NTOT IS TOTAL NO. OF BOXES + 2* NO. OF INTERFERENCE ELEMENTS
      NTOT     =  NTP + NTZ + NTY
      MTOT     =  NTOT
      NDTOB    =  0
      NDTOP    =  0
      N1  = NAA
      N2 = N1+ NB
      N3 = N2+ MAXO(NB,NP)
      N4 = N3+ NODES
      N5 = N4+ NODES
      N6 = N5+ NODES
      N8  = N6  + MSTRIP+NB
      N9 = N8+ NTOTAL
      N10 = N9+ NTOTAL
      N11 = N10+ NTOTAL
      L1  = N11+ NTOTAL
C
      CALL SPLINE (NTPH,NTPDH,NTPH4,NDTOB,NDTOP,
     1   ELXI, ELYI, ELZI, WORK(L1)
     2 , NSBEA, NBARAY, NCARAY, NSARAY, XIS1, XIS2, X, CG, SG, YS, ZS
     3 , XIC, XIJ, PHI, PHIZ, PHIY, MDOF, NODES, MODES, MDOFB, IFNEWH
     4 , WORK(N1), WORK(N2), WORK(N3), WORK(N4), WORK(N5), WORK(N6)
     5 , WORK(N8), WORK(N9), WORK(N10), WORK(N11), H, DHDX )
      NMODES(1)=          NSYM
      IF  (NB .NE. 0)  NMODES(2) = NSYM
      NMODES(3)=  NMODES(2)
      NMODES(4)=          NASYM
      IF  (NB .NE. 0)  NMODES(5) = NASYM
      NMODES(6)=  NMODES(5)
      NSLEND   =  NSBETO + NB
      NTMAX    =  MAXO(NTOT, NSLEND)
      L1 = NAA
      L2       =  L1 + NTMAX
      REWIND  MASTH
      REWIND  MASTDH
      REWIND  MASTH4
C
      CALL COTP (NTPH,NTPDH,NTPH4,MASTH,MASTDH,MASTH4, IFNEWH,
     1 NSYM,NASYM,MDOFB ,MODES,NB,NTMAX,
     2 PHIN,PHIZ,PHIY,WORK(L1),WORK(L2) )
C
      IF  (IFNEWH .EQ. 0)  GO TO  250
      NTDACF   =  34
      NROW     =  NODES+4
      MAXSEC   =  10
      MAXPR    =  10
```

```
      I1        =   NAA
      I2        =   I1 + NTP
      I3        =   I2 + NTP
      I4        =   I3 + NODES
      I5        =   I4 + NODES
      I6        =   I5 + NODES
      I7        =   I6 + NTP
      I8        =   I7 + NROW
      I9        =   I8 + MAXSEC
      I10       =   I9 + MAXPR
      I11       =   I10+ MAXPR*MAXSEC*2
      I12       =   I11+ NROW*NROW
      CALL  NEWH (NTI , NOUT, NTDACF, MASTH4, MASTH, MASTDH, NTP, NSYM,
     1      NASYM, NROW, MODES, NBARAY, NP, NCARAY, YS, ZS, WORK(I1),
     2      XIC, X, WORK(I2), H, DHDX, WORK(I3), WORK(I4), WORK(I5),
     3      ELXI, ELYI, ELZI, WORK(I6), PHIN, WORK(I7), WORK(I8),
     4      WORK(I9), WORK(I10), WORK(I11), WORK(I12), NODES, MDOFP,
     5      PHI , MDOF, NODES )
      NAA = IFNEWH
  250 CONTINUE
      IF  (NGUST .EQ. 0)  GO TO  270
      WRITE  (NOUT,120)
  252 READ (NTI,30) NN, XCOS, YCOS, ZCOS
      IF (NN.LE.0) GO TO 253
      CR(1,NN) = XCOS
      CR(2,NN) = YCOS
      CR(3,NN) = ZCOS
      GO TO 252
  253 CONTINUE
      DO  260  NR = 1, NG
      WRITE (NOUT,130) NR, (CR(MR,NR), MR=1,3)
  260 CONTINUE
C
  270 CONTINUE
      NMSYM     =   NSYM  + NG
      NMASYM    =   NASYM + NG
      NMDTOT    =   NMSYM + NMASYM
      CALL  WRAERO (4HCR   ,0,CR,NOUT,NER)
      CALL  WRAERO (4HRK   ,0,FREQ,NOUT,NER)
      REWIND  NTGF
      L1 = NAA
C
C *** START OF FREQUENCY LOOP
C
      DO  370  JKR = 1, NK
      REWIND MASTCP
      REWIND MASTFZ
      REWIND MASTFY
      KR        =   FREQ(JKR)
      IF  (JKR .GT. NKD) GO TO  362
      IF  (IOPT .EQ. 4)  GO TO  360
      IF  (IOPT .EQ. 2)  GO TO  280
```

```
      L2  = L1  + 2*NTOT                                                SDLM 310
      L3  = L2  + 2*NTOT                                                SDLM 311
      L4  = L3  + 2*NTOT                                                SDLM 312
      NAA = L4  + 2*NTOT                                                SDLM 313
C                                                                       SDLM 314
      CALL GEND (IPR3,NTAPE,NTOT,IOPT,MASTCT,WORK(L1),                  SDLM 315
     1    WORK(L1), WORK(L2), WORK(L3), WORK(L3), WORK(L4)              SDLM 316
     2 , IFLA, NBEA, NT12, NBARAY, NCARAY, YB, ZB, ARB, AVR, RIA, XLE   SDLM 317
     3 , XTE, TH1A, TH2A, X, CG, EE, SG, YS, ZS, XIC, YIN, ZIN, DELX    SDLM 318
     4 , XLAM, SCALER )                                                 SDLM 319
C                                                                       SDLM 320
      IF  (JKR .EQ. NK .AND. IOPT .EQ. 1)  END FILE MASTDT              SDLM 321
  280 CONTINUE                                                          SDLM 322
      IF  (NB .EQ. 0)  GO TO  290                                       SDLM 323
      REWIND  MASTH                                                     SDLM 324
      REWIND  MASTDH                                                    SDLM 325
      READ    (MASTH )  NSYMB, NASYMB                                   SDLM 326
      READ    (MASTDH)                                                  SDLM 327
C                                                                       SDLM 328
      CALL         SB( NMT  , NSYMB  , NASYMB, NOUT, MASTH, MASTDH,     SDLM 329
     1   IPR2, WORK(L1), H, DT, DTA                                     SDLM 330
     2 , NBEA,         NSBEA, NCARAY, NSARAY, AO, YB, ZB, ARB           SDLM 331
     3 , AOP, XIS1, XIS2, X, CG, SG, YS, ZS, CR, NG )                   SDLM 332
C                                                                       SDLM 333
      NSBMIN    =  NSBETO                                               SDLM 334
      NSBET2    =   2 * NSBETO                                          SDLM 335
      NTSBE     =  NTZS + NTYS                                          SDLM 336
      LENGTH    =  NTOT + NTSBE                                         SDLM 337
      L2  = L1  + 4*LENGTH                                              SDLM 338
C                                                                       SDLM 339
      CALL        BFSMAT(LENGTH, NE    , NB    , NP    , NTP   , NTOT  , SDLM 340
     1                   NBFM  , NBFMA , FMACH , REFC  , KR    , YB    , SDLM 341
     2            ZB   , YS    , ZS    , X     , DELX  , EE    , XIC   , SDLM 342
     3            SG   , CG    , ARB   , RIA   , AO    , XIS1  , XIS2  , SDLM 343
     4            AVR  , NSARAY, NCARAY, NSBEA , NBEA(1,1)     ,         SDLM 344
     5            NAS  , NASB  , WORK(L1), WORK(L2), SCALER )           SDLM 345
  290 CONTINUE                                                          SDLM 346
      REWIND  NWT                                                       SDLM 347
      REWIND  NWTA                                                      SDLM 348
      L2          =  L1 + 2*NTOT                                        SDLM 349
      L3          =  L2 + 2*NTOT                                        SDLM 350
C                                                                       SDLM 351
      CALL WANCWT (IPR1,LINES,NOUT,NTOT,NTP,NB,NDW,MASTH,               SDLM 352
     1            MASTDH, NWT, NWTA,                              REFC, SDLM 353
     2            KR,  H, DHDX,  WORK(L1), WORK(L2), WORK(L3)          ) SDLM 354
C                                                                       SDLM 355
      CALL        GUST(  NWT , NWTA  , NTOT, NTP , NB, NBARAY, NCARAY,  SDLM 356
     1           NBEA, FMACH, KR, REFC, X, YS, ZS, SG, CG, CR, NG,      SDLM 357
     2   IPR1, LINES, NOUT, WORK(L1), WORK(L2), NDW, NMT, NASYM, DT )   SDLM 358
C                                                                       SDLM 359
      NMSYM    = NSYM + NG                                              SDLM 360
      NMASYM   = NASYM + NG                                             SDLM 361
```

```
      IF  (NASYM .EQ. 0)  NMASYM = 0                                      SDLM 362
      NMDTOT    =  NMSYM + NMASYM                                         SDLM 363
      NWTAP     =  NWT                                                    SDLM 364
      NMODE     =  NMSYM                                                  SDLM 365
      IGO       =  1                                                      SDLM 366
      NFTP      =  NBFM                                                   SDLM 367
      NMODEB    =  NSYM                                                   SDLM 368
  300 CONTINUE                                                            SDLM 369
      NRHSTO    =  NTOT * NMODE                                           SDLM 370
      REWIND  NWTAP                                                       SDLM 371
      REWIND  NI                                                          SDLM 372
      JX2       =  L1 - 1                                                 SDLM 373
      DO  310  JJ = 1, NMODE                                              SDLM 374
      JX1       =  JX2 + 1                                                SDLM 375
      JX2 = JX2 + 2*NTOT                                                  SDLM 376
      CALL        RWREC(IREAD , NWTAP , WORK(JX1), NTOT, NUMBR, IZERO)    SDLM 377
  310 CONTINUE                                                            SDLM 378
C                                                                         SDLM 379
      IF  (IOPT .NE. 2)  REWIND  ITAPE                                    SDLM 380
      DO  340  IX = 1, NTOT                                               SDLM 381
      JM        =  0                                                      SDLM 382
      READ   (ITAPE)  (DT(II), II=1,NTOT),  (DTA(II), II=1,NTOT)          SDLM 383
      CALL        RHSIDE(NTOT, NMODE, JM, IX, MK1, MK2, NRHSTO, RHS,      SDLM 384
     1            WORK(L1)                                             )  SDLM 385
C                                                                         SDLM 386
      GO TO  (320,330),  IGO                                              SDLM 387
  320 CONTINUE                                                            SDLM 388
      WRITE  (NI)  (DT(IK), IK=1,NTOT), (RHS(JK), JK=1,NMODE)             SDLM 389
      GO TO  340                                                          SDLM 390
  330 WRITE  (NI) (DTA(IK), IK=1,NTOT), (RHS(JK), JK=1,NMODE)             SDLM 391
  340 CONTINUE                                                            SDLM 392
C                                                                         SDLM 393
      REWINC  NI                                                          SDLM 394
      REWIND  NW                                                          SDLM 395
      CALL ZEROUT (WORK(L1),NWORK,0,0)                                    SDLM 396
      NPM = NTOT + NMODE                                                  SDLM 397
      NWORK = 2*MAX0(3*NPM,NTOT*NMODE)                                    SDLM 398
      WRITE  (NOUT,80)                                                    SDLM 399
      CALL        SOLVIT(WORK(L1), WORK(L1), NTOT, NMODE, NWORK,          SDLM 400
     1 NI,NM,NO,NW,IPR3  )                                                SDLM 401
      REWIND  NW                                                          SDLM 402
      CALL        RWREC(ICOPY ,  NW, WORK(L1), NTOT, NMODE, MASTCP     )  SDLM 403
      IF  (NB .EQ. 0)  GO TO  342                                         SDLM 404
      L5 = L1 + 2*LENGTH                                                  SDLM 405
      LWORK = LENGTH                                                      SDLM 406
      CALL        MATMUL(NW, NPSTAP, NFTP, NEWBFZ, NEWBFY, LWORK, NTSBE,  SDLM 407
     1 NSBETO,LENGTH,NTOT,NMODE,NMODEB,IPR3,KR,                           SDLM 408
     2            WORK(L1), WORK(L1), WORK(L5),  WORK(L5)              )  SDLM 409
      REWIND  NEWBFZ                                                      SDLM 410
      REWINC  NEWBFY                                                      SDLM 411
      CALL        RWREC(ICOPY , NEWBFZ, WORK(L1), NSBETO, NMODE, MASTFZ)  SDLM 412
      CALL        RWREC(ICOPY , NEWBFY, WORK(L1), NSBETO, NMODE, MASTFY)  SDLM 413
```

```
  342 GO TO  (350,360),   IGO                                           SDLM 414
  350 CONTINUE                                                          SDLM 415
      IF (NMASYM.EQ.0) GO TO 360                                        SDLM 416
      IGO        =   2                                                  SDLM 417
      NWTAP      =   NWTA                                               SDLM 418
      NMODE      =   NMASYM                                             SDLM 419
      NFTP       =   NBFMA                                              SDLM 420
      NMODEB     =   NASYM                                              SDLM 421
      GO TO   300                                                       SDLM 422
C                                                                       SDLM 423
  360 CONTINUE                                                          SDLM 424
      GO TO   368                                                       SDLM 425
  362 CONTINUE                                                          SDLM 426
      IF   (FMACH .GT. 0.01)   GO TO   366                              SDLM 427
C                                                                       SDLM 428
      WRITE   (6,364)                                                   SDLM 429
  364 FORMAT   ( /// 68H   ***   PISTON THEORY NOT APPLICABLE AT THIS MACH SDLM 430
     1NUMBER    ***    STOP        /   )                                SDLM 431
      STOP                                                              SDLM 432
C                                                                       SDLM 433
  366 CONTINUE                                                          SDLM 434
      JRUN= JRUN + 1                                                    SDLM 435
      IA1 = L1                                                          SDLM 436
      IA2 = IA1 + 2*MAXO(NTP, NSBETO)                                   SDLM 437
      IA3 = IA2 + 2*MAXO(NTP, NSBETO)                                   SDLM 438
      IA4 = IA3 + 2*NSBETO                                              SDLM 439
C                                                                       SDLM 440
      CALL          PISTON(  NB    , NTP    , NSBETO, KR     , REFC   , FMACH, SDLM 441
     1              MASTH , MASTDH, MASTH4, MASTCP, MASTFZ, MASTFY, ITAPE, SDLM 442
     2              NBARAY, NCARAY, NSBEA , YB     , ZB     , AO     , AOP   , SDLM 443
     3              XIS1   , XIS2   , CG, SG, EE, YS, ZS, H , DHDX   , DT    , SDLM 444
     4              XC              ,                  JRUN,  IPR3          , SDLM 445
     5              WORK(IA1),     WORK(IA1), WORK(IA2)       , WORK(IA3)     , SDLM 446
     6              WORK(IA3),     WORK(IA4)                  , NG, CR        ) SDLM 447
C                                                                       SDLM 448
      MASTH4     =   ITAPE                                              SDLM 449
      MTOT       =   NTP                                                SDLM 450
  368 CONTINUE                                                          SDLM 451
      REWIND   MASTCP                                                   SDLM 452
      IF   (NB .NE. 0)   REWIND   MASTFZ                                SDLM 453
C                                                                       SDLM 454
      NSMAX      =   MAXO(NSTRIP, NSBETO)                               SDLM 455
      NSMAX      =   MAXO(NSMAX , NMDTOT)                               SDLM 456
      NTMAX      =   MAXO(NTOT, NSBETO)                                 SDLM 457
      L2   = L1   + 2*NTMAX                                             SDLM 458
      L3   = L2   + 2*NTMAX                                             SDLM 459
C                                                                       SDLM 460
      CALL          GENF(NMODES,MTOT,NTP,NB,   YB   ,NSBETO,IPR2,       SDLM 461
     1              LINES, NOUT, NDZ    , NDY    , MASTH  , MASTH4, NTAERO, SDLM 462
     2              NTGF, MASTCP, MASTFZ, MASTFY, JKR, H, DHDX, NASYM,  SDLM 463
     3    DELA, XIS1, XIS2, NSBEA, WORK(L1), WORK(L2), WORK(L3), NG )   SDLM 464
C                                                                       SDLM 465
```

```
      L2  = L1  + 2*NTOT                                                SDLM 466
      L3  = L2  + 2*NSMAX                                               SDLM 467
      L4  = L3  + 2*NSMAX                                               SDLM 468
      L5  = L4  + 2*NSMAX                                               SDLM 469
      L6  = L5  + 2*NSMAX                                               SDLM 470
      L7  = L6  + 2*NSMAX                                               SDLM 471
      L8  = L7  + 2*NB                                                  SDLM 472
      L9  = L8  + 2*NB                                                  SDLM 473
      L10 = L9  + 2*NB                                                  SDLM 474
      L11 = L10 + 2*NB                                                  SDLM 475
      L12 = L11 + MSTRIP                                                SDLM 476
      CALL        AERO(MFIX1,MFIX2,NG,NMSYM, MTOT,NOUT, MASTCP, MASTFZ, SDLM 477
     1          MASTFY, NSMAX,  REFC   ,    WORK(L1), WORK(L2), WORK(L3), SDLM 478
     2   WORK(L4), WORK(L5), WORK(L6)                                   SDLM 479
     2 , ISSTR,NSBEA,NBARAY,NCARAY,YB,ZB,XIS1,XIS2,CG,CS,EE,SG,YS,ZS    SDLM 480
     3 , XIC, XIJ, DELX, COORD                                          SDLM 481
     4 , WORK(L7), WORK(L8), WORK(L9), WORK(L10), WORK(L11), WORK(L12) )SDLM 482
C                                                                       SDLM 483
  370 CONTINUE                                                          SDLM 484
C                                                                       SDLM 485
C *** END OF FREQUENCY LOOP                                             SDLM 486
C                                                                       SDLM 487
      RETURN                                                            SDLM 488
      END                                                               SDLM 489
```

```
      SUBROUTINE SNPDF(SL,CL,TL,SGS,CGS,SGR,CGR,XO,YO,ZO,EE,DIJ,BETA,CV)SNPDF   2
C    ***    CALCULATES THE STEADY PART OF THE INFLUENCE COEFFICIENT     SNPDF   3
C           MATRIX ELEMENTS                                             SNPDF   4
      REAL     KR,KRDBR                                                 SNPDF   5
   10 FORMAT  (1HO, 6F15.8)                                             SNPDF   6
C                                                                       SNPDF   7
      TEST1= 0.9999                                                     SNPDF   8
      TEST2= 0.0001*EE                                                  SNPDF   9
C   ***  TEST1  AND TEST2  SERVE AS A MEASURE OF  'NEARNESS'  WITH      SNPDF  10
C   RESPECT TO THE BOUND-  AND TRAILING VORTICES RESPECTIVELY  --  SEE  SNPDF  11
C   TESTS BELOW                                                         SNPDF  12
C   ***  NOTE  THAT THE MACH NUMBER EFFECT IS ACCOUNTED FOR BY STRETCHINGSNPDF  13
C   THE  X-COORDINATES AND THE  SWEEP ANGLE OF THE BOUND VORTEX LINE    SNPDF  14
      TLB   = TL/BETA                                                   SNPDF  15
      SQTLB= SQRT(1.0+TLB**2)                                           SNPDF  16
      SLB   = TLB/SQTLB                                                 SNPDF  17
      CLB   = 1.0/SQTLB                                                 SNPDF  18
      CAVE = CV                                                         SNPDF  19
      CLSGS =  CLB*SGS                                                  SNPDF  20
      CLCGS =  CLB*CGS                                                  SNPDF  21
      EX  =EE*TLB                                                       SNPDF  22
      EY  =EE*CGS                                                       SNPDF  23
      EZ  =EE*SGS                                                       SNPDF  24
      XOB =   XO /  BETA                                                SNPDF  25
      RIX = XOB+ EX                                                     SNPDF  26
      RIY = YO + EY                                                     SNPDF  27
      RIZ = ZO + EZ                                                     SNPDF  28
      RIMAG = SQRT(RIX**2 + RIY**2 + RIZ**2)                            SNPDF  29
      ROX = XOB- EX                                                     SNPDF  30
      ROY = YO - EY                                                     SNPDF  31
      ROZ = ZO - EZ                                                     SNPDF  32
      ROMAG =  SQRT(ROX**2 + ROY**2 + ROZ**2)                           SNPDF  33
      CAB = (RIX*SLB+ RIY*CLCGS + RIZ*CLSGS)/RIMAG                      SNPDF  34
      CBB = (ROX*SLB+ ROY*CLCGS + ROZ*CLSGS)/ROMAG                      SNPDF  35
      CBI = -RIX/RIMAG                                                  SNPDF  36
      CAO =   ROX/ROMAG                                                 SNPDF  37
      RICAB = RIMAG*CAB                                                 SNPDF  38
      DBX = RIX - RICAB*SLB                                             SNPDF  39
      DBY = RIY - RICAB*CLCGS                                           SNPDF  40
      DBZ = RIZ - RICAB*CLSGS                                           SNPDF  41
      DB2 = DBX**2 + DBY**2 + DBZ**2                                    SNPDF  42
      DI2 = RIY**2 + RIZ**2                                             SNPDF  43
      DO2 = ROY**2 + ROZ**2                                             SNPDF  44
      ACAB = ABS(CAB)                                                   SNPDF  45
      ACBB = ABS(CBB)                                                   SNPDF  46
C   ***  THE FOLLOWING IS A TEST TO SEE IF THE RECEIVING POINT LIES ON  SNPDF  47
C   OR NEAR THE BOUND VORTEX  --  IF SO, THE CONTRIBUTION OF THE BOUND  SNPDF  48
C   VORTEX IS SET TO ZERO                                               SNPDF  49
      IF (ACAB.GT.TEST1)  GO TO  30                                     SNPDF  50
      IF (ACBB.GT.TEST1)  GO TO  30                                     SNPDF  51
      CACB =(CAB - CBB)/DB2                                             SNPDF  52
      GO TO  60                                                         SNPDF  53
```

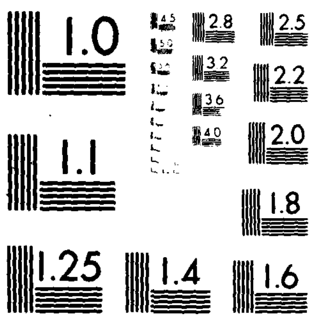
```
   30 IF (CAB*CBB)40,50,50
   40 CACB = 0.
      GO TO 60
   50 CACB = 0.5*ABS((1./RIMAG**2)-(1./ROMAG**2))
   60 CONTINUE
      VBY = CACB * (DBX*CLSGS - DBZ*SLB)
      VBZ = CACB * (DBY*SLB- DBX*CLCGS)
C  ***  TEST TO SEE IF THE RECEIVING POINT LIES ON OR NEAR THE
C  INBOARD TRAILING VORTEX  --  IF SO, THE CONTRIBUTION OF THE
C  INBOARD TRAILING VORTEX IS SET TO ZERO
      IF (DI2.GT.TEST2)  GO TO 62
      VIY  = 0.0
      VIZ  = 0.0
      GO TO  64
   62 CONTINUE
      ONECBI = (1.0 - CBI)/DI2
      VIY = ONECBI*RIZ
      VIZ = -ONECBI*RIY
   64 CONTINUE
C  ***  TEST TO SEE IF THE RECEIVING POINT LIES ON OR NEAR THE
C  OUTBOARD TRAILING VORTEX  --  IF SO, THE CONTRIBUTION OF THE
C  OUTBOARD TRAILING VORTEX IS SET TO ZERO
      IF (DO2.GT.TEST2)  GO TO 66
      VOY  = 0.0
      VOZ  = 0.0
      GO TO  68
   66 CONTINUE
      CAOONE = (1.0 + CAO)/DO2
      VOY = -CAOONE*ROZ
      VOZ = CAOONE*ROY
   68 CONTINUE
      VX = DBZ*CLCGS - DBY*CLSGS
      VY = VBY + VIY + VOY
      VZ = VBZ + VIZ + VOZ
      WW= VY*SGR - VZ*CGR
      DIJ = WW* CAVE / 25.132741
   70 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SOLVIT(A    ,RA,ND,MD,KD,NI,NM,NO,NW,NPR1)
C
C  ***    SOLVES SYSTEMS OF SIMULTANEOUS LINEAR EQUATIONS   OF
C         ND   UNKNOWNS AND  MD  RIGHT HAND SIDES
C
C           ****    ***/  *******  *****  *       ***/          *
C           *   *  *  /*     *     *      *      *  /*       ******
C     ****  ****   * / *     *     ***    *      * / *  ****    *  *
C           *      */  *     *     *      *      */  *          ***
C           *      /***      *     *      *****  /***           *  *
C
C                             - C O M P L E X -
C            D I R E C T   M A T R I X   S O L U T I O N
C
C     WRITTEN BY J. L. HESS * PROGRAMMED BY T. M. RIDDELL
C
      COMPLEX     A( KD )
C
      DIMENSION  RA(2, KD)
C
C
      LOGICAL  LAST
C
    1  FORMAT (4H1THE I4, 2H X I3, 15H MATRIX EXCEEDS I6, 7H WORDS.)
    2 FORMAT ( 6E16.6 )
    3 FORMAT ( //12H  COLUMN NO.,I6,2X,17HOF GAMMAS FOLLOWS,/ )
    4 FORMAT  ( 1H1 /// )
      N = ND
      M = MD
      MM= NM
      KORE = KD/2
      NPM = N + M
      IF (MAX0(3 * NPM, M * N) .LE. KORE) GO TO 5
      WRITE (6,  1 ) NQUAD, NCFLG, KORE
      STOP
    5 MT = MM
      REWIND MT
      NIN = NI
      REWIND NIN
      NOUT = NO
      REWIND NOUT
      MP1 = M + 1
      NN = N
      NEL = NPM
C
C - - CALCULATE THE MAXIMUM NO. OF ROWS, 'K'
C
   10 K = (KORE - NEL) / NEL
C
C - - TEST TO SEE IF THE REST OF THE MATRIX WILL FIT IN CORE
C
      LAST = K .GE. NN
```

AD-A106 520 DOUGLAS AIRCRAFT CO . LONG BEACH CA F/G 18/3
NUCLEAR BLAST RESPONSE COMPUTER PROGRAM. VOLUME III. PROGRAM LI--ETC(U)
AUG 81 J A MCGREW, H H CROXEN, T P KALMAN DNA001-75-C-0216

UNCLASSIFIED AFWL-TR-81-32-VOL-3 NL

3 OF 5
A
106520



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

```
      IF (LAST) K = NN                                                  SOLVIT54
C                                                                       SOLVIT55
C - - READ 'K' ROWS OF THE AUGMENTED 'A' MATRIX                          SOLVIT56
C                                                                       SOLVIT57
   30 NT = 0                                                            SOLVIT58
      DO 40 IB = 1, K                                                   SOLVIT59
      NS = NT + 1                                                       SOLVIT60
      NT = NT + NEL                                                     SOLVIT61
   40 READ (NIN) (A(IO), IO = NS, NT)                                   SOLVIT62
C                                                                       SOLVIT63
C - - CHECK TO SEE IF WE WERE UNLUCKY ENOUGH TO END UP WITH ONLY ONE ROWSOLVIT64
C                                                                       SOLVIT65
      IF (K .EQ. 1) GO TO 90                                            SOLVIT66
C                                                                       SOLVIT67
C - - 'K' IS GREATER THAN '1' SO WE CAN START THE TRIANGULARIZATION     SOLVIT68
C                                                                       SOLVIT69
      NELP1 = NEL + 1                                                   SOLVIT70
      NS = - NEL                                                        SOLVIT71
      NELP2 = NELP1 + 1                                                 SOLVIT72
C                                                                       SOLVIT73
C - - FORM THE 'TRAPEZOIDAL' ARRAY  (8)                                 SOLVIT74
C                                                                       SOLVIT75
      DO 50 IB = 2, K                                                   SOLVIT76
      NP = NELP2 - IB                                                   SOLVIT77
      NS = NS + NELP1                                                   SOLVIT78
      NT = NS                                                           SOLVIT79
      DO 50 IO = IB, K                                                  SOLVIT80
      NT = NT + NEL                                                     SOLVIT81
      MN = NT                                                           SOLVIT82
      NB = NS                                                           SOLVIT83
C                                                                       SOLVIT84
C     A(NT) = -A(NT) / A(NS)                                            SOLVIT85
C                                                                       SOLVIT86
      TEMPR = REAL(A(NT))                                               SOLVIT87
      DENOM = - ( REAL(A(NS)) ** 2 + AIMAG(A(NS)) ** 2 )                SOLVIT88
      RA(1,NT) = (TEMPR        * REAL(A(NS)) + AIMAG(A(NT)) *           SOLVIT89
     *            AIMAG(A(NS))) / DENOM                                 SOLVIT90
      RA(2,NT) = (AIMAG(A(NT)) * REAL(A(NS)) - TEMPR *                  SOLVIT91
     *            AIMAG(A(NS))) / DENOM                                 SOLVIT92
C                                                                       SOLVIT93
      DO 50 NF = 2, NP                                                  SOLVIT94
      MN = MN + 1                                                       SOLVIT95
      NB = NB + 1                                                       SOLVIT96
C                                                                       SOLVIT97
C  50 A(MN) = A(MN) + A(NT) * A(NB)                                     SOLVIT98
C                                                                       SOLVIT99
      RA(1,MN) = RA(1,MN) + REAL(A(NT)) * REAL(A(NB)) -                 SOLVI100
     *            AIMAG(A(NT)) * AIMAG(A(NB))                           SOLVI101
C                                                                       SOLVI102
   50 RA(2,MN) = RA(2,MN) + REAL(A(NT)) * AIMAG(A(NB)) +                SOLVI103
     *            AIMAG(A(NT)) * REAL(A(NB))                            SOLVI104
C                                                                       SOLVI105
```

```
      IF (LAST) GO TO 90
C
C - - WRITE THE 'TRAPEZOIDAL' MATRIX ON TAPE
C
      NT = 0
      NP = NEL
      NS = - NEL
      DO 60 IO = 1, K
      NS = NS + NELP1
      NT = NT + NEL
      WRITE (MT) NP, (A(IB), IB = NS, NT)
   60 NP = NP - 1
      NP = NP - M
      NS = KORE - NEL + 1
C
C - - READ ANOTHER ROW
C
      DO 80 IO = 1, NP
      READ (NIN) (A(IB), IB = NS, KORE)
C
C - - MODIFY THIS ROW BY THE 'TRAPEZOIDAL' ARRAY
C
      NT = 1
      MN = NS
      DO 70 IB = 1, K
      NB = NT
      NF = MN + 1
C
C     A(MN) = -A(MN) / A'NT)
C
      TEMPR = REAL(A(MN))
      DENOM = - (REAL(A(NT)) ** 2 + AIMAG(A(NT)) ** 2)
      RA(1,MN) = (TEMPR        * REAL(A(NT)) + AIMAG(A(MN)) *
     *            AIMAG(A(NT))) / DENOM
      RA(2,MN) = (AIMAG(A(MN)) * REAL(A(NT)) - TEMPR *
     *            AIMAG(A(NT))) / DENOM
C
      DO 65 NN = NF, KORE
      NB = NB + 1
C
C  65 A(NN) = A(NN) + A(MN) * A(NB)
C
      RA(1,NN) = RA(1,NN) + REAL(A(MN)) * REAL(A(NB)) -
     *           AIMAG(A(MN)) * AIMAG(A(NB))
C
   65 RA(2,NN) = RA(2,NN) + REAL(A(MN)) * AIMAG(A(NB)) +
     *           AIMAG(A(MN)) * REAL(A(NB))
C
      MN = NF
   70 NT = NT + NELP1
C
C - - WRITE THE MODIFIED ROW ON TAPE
```

```
C                                                                       SOLVI158
   80 WRITE (NOUT)     (A(NT), NT = MN, KORE)                           SOLVI159
      REWIND NOUT                                                       SOLVI160
      REWIND NIN                                                        SOLVI161
C                                                                       SOLVI162
C - - SWITCH THE TAPES                                                  SOLVI163
C                                                                       SOLVI164
      NT = NIN                                                          SOLVI165
      NIN = NOUT                                                        SOLVI166
      NOUT = NT                                                         SOLVI167
C                                                                       SOLVI168
C - - RE-CALCULATE ROW LENGTH AND LOOP BACK                             SOLVI169
C                                                                       SOLVI170
      NEL = NEL - K                                                     SOLVI171
      NN = NEL - M                                                      SOLVI172
      GO TO 10                                                          SOLVI173
C                                                                       SOLVI174
C - - REWIND ALL TAPES                                                  SOLVI175
C                                                                       SOLVI176
   90 REWIND  MT                                                        SOLVI177
      REWIND NIN                                                        SOLVI178
      REWIND NOUT                                                       SOLVI179
C                                                                       SOLVI180
C - - CONDENSE THE MATRIX                                               SOLVI181
C                                                                       SOLVI182
      NN = NEL                                                          SOLVI183
      NL    = NEL + 1                                                   SOLVI184
      IF (K .EQ. 1) GO TO 105                                           SOLVI185
      NS = 1                                                            SOLVI186
      NT = NEL                                                          SOLVI187
      DO 100 IB = 2, K                                                  SOLVI188
      NS = NS + NELP1                                                   SOLVI189
      NT = NT + NEL                                                     SOLVI190
      DO 100 IO = NS, NT                                                SOLVI191
      A(NL) = A(IO)                                                     SOLVI192
  100 NL    = NL + 1                                                    SOLVI193
  105 NI      KORE - K * M + 1                                          SOLVI194
C                                                                       SOLVI195
C - - THERE, NOW WE CAN START THE BACK-SOLUTION                         SOLVI196
C * * NOTE..THE FIRST AVAILABLE LOCATION FOR THE SOLUTIONS IS A(NI)     SOLVI197
C                                                                       SOLVI198
      NREM = N                                                          SOLVI199
      NEL = NPM                                                         SOLVI200
      LAST = K .EQ. N                                                   SOLVI201
      NPASS = 0                                                         SOLVI202
C                                                                       SOLVI203
C - - SOLVE FOR THE ANSWERS CORRESPONDING TO 'K' ROWS                   SOLVI204
C                                                                       SOLVI205
  110 KM1 = K - 1                                                       SOLVI206
      KP1 = K + 1                                                       SOLVI207
      NS = NL - MP1                                                     SOLVI208
      NPASS = NPASS + 1                                                 SOLVI209
```

```
      DO 130 MN = 1, M                                                  SOLVI210
      NF = NS + MN                                                      SOLVI211
C                                                                       SOLVI212
C     A(NF) = A(NF) / A(NS)                                             SOLVI213
C                                                                       SOLVI214
      TEMPR = REAL(A(NF))                                               SOLVI215
      DENOM = REAL(A(NS)) ** 2 + AIMAG(A(NS)) ** 2                      SOLVI216
      RA(1,NF) = (TEMPR       * REAL(A(NS)) + AIMAG(A(NF)) *            SOLVI217
     *            AIMAG(A(NS))) / DENOM                                 SOLVI218
      RA(2,NF) = (AIMAG(A(NF)) * REAL(A(NS)) - TEMPR *                  SOLVI219
     *            AIMAG(A(NS))) / DENOM                                 SOLVI220
C                                                                       SOLVI221
      NT = NS                                                           SOLVI222
      IF (KM1 .EQ. 0) GO  TO 130                                        SOLVI223
      DO 125 IB = 1, KM1                                                SOLVI224
      NF = NF - IB - M                                                  SOLVI225
      NT = NT - MP1 - IB                                                SOLVI226
C                                                                       SOLVI227
C     SUM = 0.0                                                         SOLVI228
C                                                                       SOLVI229
      SUMR = 0.0                                                        SOLVI230
      SUMI = 0.0                                                        SOLVI231
C                                                                       SOLVI232
      NP = NF                                                           SOLVI233
      N2 = MP1 + IB                                                     SOLVI234
      DO 120 IO = 1, IB                                                 SOLVI235
      NN = NT + IO                                                      SOLVI236
      NP = NP + N2 - IO                                                 SOLVI237
C                                                                       SOLVI238
C 120 SUM = SUM + A(NN) * A(NP)                                         SOLVI239
C                                                                       SOLVI240
      SUMR = SUMR + REAL(A(NN)) * REAL(A(NP)) - AIMAG(A(NN)) *          SOLVI241
     *        AIMAG(A(NP))                                              SOLVI242
C                                                                       SOLVI243
  120 SUMI = SUMI + REAL(A(NN)) * AIMAG(A(NP)) + AIMAG(A(NN)) *         SOLVI244
     *        REAL(A(NP))                                               SOLVI245
C                                                                       SOLVI246
C 125 A(NF) = (A(NF) - SUM) / A(NT)                                     SOLVI247
C                                                                       SOLVI248
      DENOM = REAL(A(NT)) ** 2 + AIMAG(A(NT)) ** 2                      SOLVI249
C                                                                       SOLVI250
      TEMPR = REAL(A(NF)) - SUMR                                        SOLVI251
      TEMPI = AIMAG(A(NF)) - SUMI                                       SOLVI252
C                                                                       SOLVI253
      RA(1,NF) = (TEMPR * REAL(A(NT)) + TEMPI * AIMAG(A(NT))) / DENOM   SOLVI254
  125 RA(2,NF) = (TEMPI * REAL(A(NT)) - TEMPR * AIMAG(A(NT))) / DENOM   SOLVI255
C                                                                       SOLVI256
  130 CONTINUE                                                          SOLVI257
C                                                                       SOLVI258
C - - MOVE THE SOLUTIONS TO CONTIGUOUS LOCATIONS STARTING AT A(N1)      SOLVI259
C                                                                       SOLVI260
      N1 = KORE + 1                                                     SOLVI261
```

```
      DO 140 NN = 1, K                                                  SOLVI262
      DO 135 MN = 1, M                                                  SOLVI263
      NL = NL - 1                                                       SOLVI264
      N1 = N1 - 1                                                       SOLVI265
  135 A(N1) = A(NL)                                                     SOLVI266
  140 NL = NL - NN                                                      SOLVI267
C                                                                       SOLVI268
C - - WRITE THE SOLUTIONS ON TAPE                                       SOLVI269
C                                                                       SOLVI270
      WRITE (NIN) K                                                     SOLVI271
      NS = N1 - 1                                                       SOLVI272
      DO 145 MN = 1, M                                                  SOLVI273
      NT = NS + MN                                                      SOLVI274
  145 WRITE ( NIN ) (A(IO), IO = NT, KORE, M)                           SOLVI275
C                                                                       SOLVI276
C - - TEST IF THIS IS THE LAST PASS                                     SOLVI277
C                                                                       SOLVI278
      IF (LAST) GO TO 200                                               SOLVI279
C                                                                       SOLVI280
C - - WE MUST NOW MODIFY THE TRIANGULAR MATRIX TO REFLECT THE EFFECT CF SOLVI281
C     THE SOLUTIONS OBTAINED SO FAR  (EQ 21)                            SOLVI282
C * * NOTE..LOCATIONS A(1) TO A(N1-1) ARE NOW FREE TO USE               SOLVI283
C                                                                       SOLVI284
C - - CALCULATE THE NEXT VALUES OF 'NEL' AND 'NREM'                     SOLVI285
C                                                                       SOLVI286
      NELOLD = NEL                                                      SOLVI287
      KOLD = K                                                          SOLVI288
      NEL = NEL - K                                                     SOLVI289
      NREM = NREM - K                                                   SOLVI290
C                                                                       SOLVI291
C - - NOW APPLY THE INCREDIBLE FORMULA FOR THE NEW 'K'                  SOLVI292
C                                                                       SOLVI293
      K = (-4 * M - 1) / 2 + IFIX(SQRT(0.25 + FLOAT((4 * M + 2) * M +   SOLVI294
     1 2 * (KORE - NELOLD))))                                           SOLVI295
      NROW = NREM - K + 1                                               SOLVI296
      IF (K .LT. NREM) GO TO 150                                        SOLVI297
      LAST = .TRUE.                                                     SOLVI298
      NROW = 1                                                          SOLVI299
      K = NREM                                                          SOLVI300
  150 NS = 1                                                            SOLVI301
      NT = NELOLD + 1                                                   SOLVI302
C                                                                       SOLVI303
C - - READ IN THE ROWS TO BE MODIFIED                                   SOLVI304
C                                                                       SOLVI305
      DO 190 IB = 1, NREM                                               SOLVI306
      NT = NT - 1                                                       SOLVI307
      IF (IB .LE. NROW) GO TO 160                                       SOLVI308
      NS = NS + NN                                                      SOLVI309
      NT = NT + NN                                                      SOLVI310
  160 READ ( MT ) NN, (A(IO), IO = NS, NT)                              SOLVI311
      NP = N1 - 1                                                       SOLVI312
      NF = NT - M - KM1                                                 SOLVI313
```

```
      NN = NN - KOLD                                                    SOLVI314
      DO 170 MN = 1, M                                                  SOLVI315
      N2 = NF                                                           SOLVI316
      NA = NP + MN                                                      SOLVI317
      NB = NA                                                           SOLVI318
C                                                                       SOLVI319
C     SUM = 0.0                                                         SOLVI320
C                                                                       SOLVI321
      SUMR = 0.0                                                        SOLVI322
      SUMI = 0.0                                                        SOLVI323
C                                                                       SOLVI324
      DO 165 IO = 1, KOLD                                               SOLVI325
C                                                                       SOLVI326
C     SUM = SUM + A(N2) * A(NA)                                         SOLVI327
C                                                                       SOLVI328
      SUMR = SUMR + REAL(A(N2)) * REAL(A(NA)) - AIMAG(A(N2)) *          SOLVI329
     *       AIMAG(A(NA))                                               SOLVI330
      SUMI = SUMI + REAL(A(N2)) * AIMAG(A(NA)) + AIMAG(A(N2)) *         SOLVI331
     *       REAL(A(NA))                                                SOLVI332
C                                                                       SOLVI333
      N2 = N2 + 1                                                       SOLVI334
  165 NA = NA + M                                                       SOLVI335
      N2 = N2 + MN - 1                                                  SOLVI336
C                                                                       SOLVI337
C 170 A(N2) = A(N2) - SUM                                               SOLVI338
C                                                                       SOLVI339
      RA(1,N2) = RA(1,N2) - SUMR                                        SOLVI340
  170 RA(2,N2) = RA(2,N2) - SUMI                                        SOLVI341
C                                                                       SOLVI342
C - - WRITE THE MODIFIED ROW ON TAPE OR CONDENSE THE ROW                SOLVI343
C                                                                       SOLVI344
      NL = NT - M + 1                                                   SOLVI345
      IF (IB .GE. NROW) GO TO 175                                       SOLVI346
      NF = NL - KP1                                                     SOLVI347
      WRITE (NOUT) NN, (A(IO), IO = NS, NF), (A(IO), IO = NL, NT)       SOLVI348
      GO TO 190                                                         SOLVI349
  175 NF = NL - KOLD                                                    SOLVI350
      DO 180 MN = NL, NT                                                SOLVI351
      A(NF) = A(MN)                                                     SOLVI352
  180 NF = NF + 1                                                       SOLVI353
  190 CONTINUE                                                          SOLVI354
      REWIND MT                                                         SOLVI355
      REWIND NOUT                                                       SOLVI356
C                                                                       SOLVI357
C - - SWITCH THE TAPES                                                  SOLVI358
C                                                                       SOLVI359
      NT = MT                                                           SOLVI360
      MT = NOUT                                                         SOLVI361
      NOUT = NT                                                         SOLVI362
C                                                                       SOLVI363
C - - LOOP BACK THRU THE SOLUTION                                       SOLVI364
C                                                                       SOLVI365
```

```
      NL = NF                                                            SOLVI366
      GO TO 110                                                          SOLVI367
C                                                                        SOLVI368
C - - START TO WRAP IT UP                                                SOLVI369
C                                                                        SOLVI370
  200 END FILE NIN                                                       SOLVI371
      REWIND NIN                                                         SOLVI372
      N2 = N                                                             SOLVI373
C                                                                        SOLVI374
C * * NOTE.. AT THIS POINT ALL LOCATIONS A(1) THRU A(KORE) ARE FREE      SOLVI375
C                                                                        SOLVI376
      DO 220 IB = 1, NPASS                                               SOLVI377
      READ (NIN) K                                                       SOLVI378
      N1 = N2 - K + 1                                                    SOLVI379
      NS = N1                                                            SOLVI380
      NT = N2                                                            SOLVI381
C                                                                        SOLVI382
C - - READ IN THE SOLUTIONS                                               SOLVI383
C                                                                        SOLVI384
      DO 210 IC = 1, M                                                   SOLVI385
      READ (NIN) (A(NN), NN = NS, NT)                                    SOLVI386
      NT = NT + N                                                        SOLVI387
  210 NS = NS + N                                                        SOLVI388
  220 N2 = N1 - 1                                                        SOLVI389
C                                                                        SOLVI390
C - - WRITE THE SOLUTIONS ON TAPE                                         SOLVI391
C                                                                        SOLVI392
      NT = 0                                                             SOLVI393
      LINES  = 0                                                         SOLVI394
      DO 230 IO = 1, M                                                   SOLVI395
      NS = NT + 1                                                        SOLVI396
      NT = NT + N                                                        SOLVI397
      IF (NPR1.EQ.0) GO TO 230                                           SOLVI398
      LINES  = LINES + ND/3 + 1                                          SOLVI399
      WRITE (6,3) IO                                                     SOLVI400
      WRITE (6,2) (A(NN), NN = NS, NT)                                   SOLVI401
      IF  (LINES .LT. 25)  GO TO  230                                    SOLVI402
      WRITE  (6,4)                                                       SOLVI403
      LINES    = 0                                                       SOLVI404
  230 WRITE (NW) (A(NN), NN = NS, NT)                                    SOLVI405
C                                                                        SOLVI406
      RETURN                                                             SOLVI407
      END                                                                SOLVI408
```

```
      SUBROUTINE SORT(IF1,IF2,NSUP,INSUP,I1,I2,NXQ,LMAX,X1L,X2L,X3L,YL,
     *               XP,YP,H ,DHDX,   ISP, NSBP ,NBCUM,NPCUM,NTPH,NTPDH,
     3               NTPSH,KPRINT,NTP6,KCUM1,KCUM2,KCUM3,NTOT,ISORT,NODE,
     3               SUPRH, SUPRDH )
      DIMENSION   INSUP(200),X1L(LMAX),X2L(LMAX),X3L(LMAX), YL(LMAX),
     *               XP(LMAX),     YP(LMAX), NBCUM(50), NPCUM(200)
      DIMENSION H(1), DHDX(1), NODE(1), SUPRH(1), SUPRDH(1)
C
C    ***   ISORT IS A FLAG THAT ACTIVATES EITHER  SEGMENT  1  (ISORT=1)
C                OR  SEGMENT  2  (ISORT=2)  OF THE SUBROUTINE
C
      GO TO  ( 70, 270),   ISORT
   70 CONTINUE
C
C   ------------------------------------------------------------------
C
C    ***   SEGMENT  1 EXTRACTS THE   XP-  AND YP-  ARRAYS  FOR THE
C                CURRENT  SUPERPANEL(BODY)  FROM THE COMPLETE GEOMETRY
C                ARRAYS  SUPPLIED BY  SUBROUTINE  DATA
C
      DO  90  I=1,LMAX
      XP(I)   = 0.0
      YP(I)   = 0.0
   90 CONTINUE
      I       = 0
      DO  200  J=1,NSUP
      IF  (IF1. EQ . 0)  GO TO  110
      NBODY   = INSUP(J)
      NBM1    = NBODY - 1
      IF  (NBM1. EQ . 0)  GO TO   95
      L1      = NBCUM(NBM1) + 1
      GO TO  100
   95 L1      = 1
  100 L2      = NBCUM(NBODY)
      GO TO  130
  110 CONTINUE
      NOPAN   = INSUP(J)
      NPM1    = NOPAN - 1
      IF  (NPM1. EQ . 0)  GO TO  115
      L1      = NPCUM(NPM1) + 1
      GO TO  120
  115 L1      = 1
  120 L2      = NPCUM(NOPAN)
  130 CONTINUE
      DO  180  L=L1,L2
      I       = I + 1
      IF  (IF1. EQ . 0)  GO TO  170
      XP(I)   = X3L(L)
      YP(L)   = 0.0
      GO TO  180
  170 CONTINUE
      YP(I)   =  YL(L)
```

```
      IF  (IF2 . EQ . 1)  GO TO  175
      XP(I)   = X1L(L)
      GO TO  180
  175 XP(I)   = X2L(L)
  180 CONTINUE
  200 CONTINUE
      I2      = I
      RETURN
C
C    -----------------------------------------------------------------
C
  270 CONTINUE
C
C   ***    SEGMENT 2  EXPANDS THE   H-  AND  DHDX-  COLUMNS GENERATED BY
C                SUBROUTINE  SPL2  FOR ONE SUPERPANEL(BODY)  INTO FULL
C                COLUMNS  -SUPRH-  AND  -SUPRDH-  , AND SAVES THEM ON
C                TAPE  UNITS   NTPSH  AND  NTPSDH  RESPECTIVELY.
C
      DO  500  K=1,NXQ
      READ   (NTPH)    (H(I),   I=I1,I2)
      IF  (IF1 . EQ . 0 . AND . IF2 . EQ . 1)  GO TO  275
      KCUM1   = KCUM1 + 1
      IF  (IF2  . NE . 0)  GO TO  280
      READ  (NTPDH)  (DHDX(I),   I=I1,I2)
      KCUM2   = KCUM2 + 1
      GO TO  280
  275 CONTINUE
      KCUM3   = KCUM3 + 1
  280 CONTINUE
      DO  290  L=1,LMAX
      SUPRH(L)  = 0.0
  290 SUPRDH(L) = 0.0
      IH      = 0
      L1      = 1
      L2      = 1
      DO  400  J=1,NSUP
      IF  (IF1 . EQ . 0)  GO TO  310
      NBODY   = INSUP(J)
      NBM1    = NBODY - 1
      IF  (NBM1 . EQ . 0)  GO TO  305
      L1      = NBCUM(NBM1) + 1
      GO TO  300
  305 L1      = 1
  300 L2      = NBCUM(NBODY)
      GO TO  330
  310 CONTINUE
      NOPAN   = INSUP(J)
      NPM1    = NOPAN - 1
      IF  (NPM1 . EQ . 0)  GO TO  315
      L1      = NPCUM(NPM1) + 1
      GO TO  320
  315 L1      = 1
```

```
320 L2       = NPCUM(NOPAN)                                                  SORT 106
330 CONTINUE                                                                 SORT 107
    DO   380   L=L1,L2                                                       SORT 108
    IH       = IH + 1                                                        SORT 109
    SUPRH(L)   = H(IH)                                                       SORT 110
    IF   (IF2 . NE . 0)   GO TO   380                                        SORT 111
    SUPRDH(L) = DHDX(IH)                                                     SORT 112
380 CONTINUE                                                                 SORT 113
400 CONTINUE                                                                 SORT 114
    WRITE   (NTPSH)  NODE(K), NTOT, (SUPRH (I), I=1, NTOT)                   SORT 115
    IF   (IF2 .NF. 0)   GO TO   500                                          SORT 116
    WRITE   (NTPSH)  NODE(K), NTOT, (SUPRDH(I), I=1, NTOT)                   SORT 117
500 CONTINUE                                                                 SORT 118
    RETURN                                                                   SORT 119
    END                                                                      SORT 120
```

```
      SUBROUTINE SPLINE (MASTH,MASTOH,MASTH4,NMODEB,NMODE,              SPLINE 2
     1   ELXI, ELYI, ELZI, WORK                                         SPLINE 3
     2 , NSBEA, NBARAY, NCARAY, NSARAY, XIS1, XIS2, X, CG, SG, YS, ZS   SPLINE 4
     3 , XIC, XIJ, PHI, PHIZ, PHIY, MDOF, MNODE, NMODES, NDOFB, IFNEWH  SPLINE 5
     4 , NBCUM, INSUP, NODE, XIQ, ETAQ, CSG                             SPLINE 6
     5 , X3L, YL, XP, YP, H, DHDX )                                     SPLINE 7
      DIMENSION ELXI(1), ELYI(1), ELZI(1)                               SPLINE 8
      DIMENSION WORK(1)                                                 SPLINE 9
      DIMENSION NSBEA(1), NBARAY(1), NCARAY(1), NSARAY(1), XIS1(1)      SPLINE10
      DIMENSION XIS2(1), X(1), CG(1), SG(1), YS(1), ZS(1), XIC(1),XIJ(1)SPLINE11
C    ***   SURFACE SPLINE PROGRAM                                       SPLINE12
C                                                                       SPLINE13
C                                                                       SPLINE14
      COMMON /AROCOM/ NTI, MODES                                        AROCOM 2
     X , NP, MSTRIP, NSMAX, NCMAX, NTOTAL, NB, MSBE, MBE                AROCOM 3
     Y , ND, NE, NBY, NBZ, NTO, NTP, NTY, NTZ                           AROCOM 4
     1 , NTYS, NTZS, MAXGR, MAXSTR, NSBETO, NSTRIP, KR, XM, REFA, REFC  AROCOM 5
     2 , REFS, FMACH, LINES                                             AROCOM 6
C                                                                       SPLINE16
      DIMENSION NBCUM(1), INSUP(1), NODE(1)                             SPLINE17
      DIMENSION XIQ(1), ETAQ(1), CSG(1)                                 SPLINE18
      DIMENSION X3L(1), YL(1), XP(1), YP(1), H(1), DHDX(1)              SPLINE19
      DIMENSION  PHI(MDOF,MNODE,NMODES), PHIZ(NDOFB,1), PHIY(NDOFB,1)   SPLINE20
      DATA       NTP1,NTP2,NTP3,NTP4,NTP5,NTP6,NTP9/1,2,3,4,5,6,9/      SPLINE21
      DATA       NTPH, NTPDH, NTPSH, NTPSH4 / 10, 11,  3,  8 /          SPLINE22
      DATA       KODE  / -1 /                                           SPLINE23
   10 FORMAT (6I12)                                                     SPLINE24
   20 FORMAT  ( 1H1 /// 10X, 22H***   SEGMENT  2   ***  ///  5X,        SPLINE25
     1         62HGENERATE  -H-  AND  -DHDX-  ARRAYS FOR ALL DEGREES OF SPLINE26
     2FREEDOM  //  )                                                    SPLINE27
   50 FORMAT  (1H0 /   6H    THE, I4, 26H  BODIES OF SUPERBODY  NO., I4/) SPLINE28
   60 FORMAT  (1H0 /   6H    THE, I4, 26H  PANELS OF SUPERPANEL NO., I4/) SPLINE29
   70 FORMAT  (   20I5 )                                                SPLINE30
   80 FORMAT  ( // 61H   SUPER    INDEX     MODE      NODE      NODAL POINT SPLINE31
     1COORDINATES / 8H    PANEL, 22X,3H(I),6X, 21HXI(I)          ETA(I)/)SPLINE32
   90 FORMAT  ( // 61H   SUPER    INDEX     MODE      NODE      NODAL POINT SPLINE33
     1COORDINATES / 8H     BODY, 22X,3H(I),6X, 21HXI(I)          ETA(I)/)SPLINE34
  120 FORMAT  (1H0 / 5X,I4,33H    XP-ELEMENTS FOR SUPERPANEL NO., I4/)  SPLINE35
  130 FORMAT  (1H0 / 5X,I4,33H    YP-ELEMENTS FOR SUPERPANEL NO., I4/)  SPLINE36
  140 FORMAT  (1H0 / 5X,I4,33H    XP-ELEMENTS FOR SUPERBODY  NO., I4/)  SPLINE37
  150 FORMAT  ( 6E16.6 )                                                SPLINE38
  160 FORMAT  ( 4I8, 4F15.6 )                                           SPLINE39
      IF  (IFNEWH .EQ. 0)  GO TO  180                                   SPLINE40
      DO  172   J = 1, NMODES                                           SPLINE41
      DO  170   I = 1, NDOFB                                            SPLINE42
      PHIZ(I,J)=  0.0                                                   SPLINE43
      PHIY(I,J)=  0.0                                                   SPLINE44
  170 CONTINUE                                                          SPLINE45
  172 CONTINUE                                                          SPLINE46
  180 CONTINUE                                                          SPLINE47
      K        =  0                                                     SPLINE48
C                                                                       SPLINE49
```

```
      MODE    =  0
      READ (NTI,10) NSB,NSP,NMAX,KPRINT
      IF  (NMAX .EQ. 0)  NMAX = MAXO(NTU, NSBETO)
      NMAX = NMAX+3
      NLOC      =  1
      NEXT      =  NMAX *NMAX + 1
      NEXT1 = MAXO(NTP,NSBETO+NB) + 1
      REWIND   NTP1
      REWIND   NTP2
      REWIND   NTP3
      REWIND   NTP4
      REWIND  MASTH
      REWIND  MASTDH
      REWIND  MASTH4
      LMAX = NTOTAL
      NTOT    = 0
      KCUM1   = 0
      KCUM2   = 0
      KCUM3   = 0
      L       = 0
      LX        =  0
      NBEOLD    =  0
      IF  (NB. EQ . 0)  GO TO  230
      DO  220  IB=1,NB
      NBEL      =  NSBEA(IB) + 1
      DO  210  IL=1,NBEL
      L         =  L + 1
      IF  (IL . EQ . NBEL)  GO TO  190
      LX        =  LX + 1
      X3L(L)    =  XIS1(LX)
      GO TO  200
  190 CONTINUE
C
      X3L(L)    =  XIS2(LX)
  200 CONTINUE
  210 CONTINUE
      NBEOLD = NBEOLD + NBEL
      NBCUM(IB) = NBEOLD
  220 CONTINUE
  230 CONTINUE
      SGP     = 1.0
      CGP     = 1.0
      L       = 0
      KS      = 0
      L2      =  0
      DO 280  IP=1,NP
      NTPH    = NTP4
      NC      =  NCARAY(IP)
      NS      =  NSARAY(IP)
      DO  270  N=1,NS
      KS      = KS + 1
      L1      =  L2 + 1
```

```
      L2       =   L2 + NC
      DO 260   L = L1, L2
  240 CGP      = CG(KS)
      IF   (ABS(CGP) . LE . 0.00001)   GO TO  250
      CSG(IP)= CGP
      YL(L)    =   YS(KS) / CGP
      GO TO 260
  250 SGP      = SG(KS)
      CSG(IP)= SGP
      IF   (ABS(SGP) . LE . 0.00001)   SGP = 1.0
C
C
      YL(L)    =   ZS(KS) / SGP
  260 CONTINUE
  270 CONTINUE
  280 CONTINUE
      REWIND   NTPSH
      REWIND   NTPSH4
      WRITE   (NTP6,20)
      NMODE    = 0
C
      IMDONE = 1
      IF   (NSB . EQ . 0)   GO TO  510
      NSBP     = NSB
      NTOT     = NBCUM(NB)
  300 CONTINUE
      ID          =  0
      DO   490   ISP = 1,NSBP
      IYCON       =  0
      IF2         = 0
      READ (NTI,10) IF1, NXQ, NSUP, IXCON, NACELL
      IF   (IXCON .EQ. 0)  IYCON = 1
      READ (NTI,10) (INSUP(I), I=1,NSUP)
      IF   (IF1 .NE. 1)   GO TO  340
      WRITE   (NTP6,50)  NSUP, ISP
C
  340 CONTINUE
      READ (NTI,10) (NODE(I), I=1,NXQ)
      IF   (IFNEWH .EQ. 0)   GO TO  350
      DO   342   I = 1, NXQ
      K           =   K + 1
      L           =   NODE(I)
      DO   346   N = 1, NMODES
      PHIZ(K,N)=   PHI(3, L, N)
      PHIY(K,N)=   PHI(2, L, N)
  346 CONTINUE
      IF   (NACELL. EQ. 0)   GO TO  342
      K           =   K + 1
      DO   348   N = 1, NMODES
      PHIZ(K,N)=   PHI(5, L, N)
      PHIY(K,N)=   PHI(6, L, N)
  348 CONTINUE
```

```
  342 CONTINUE                                                          SPLIN154
  350 CONTINUE                                                          SPLIN155
C                                                                       SPLIN156
      IF   (IF1 .EQ. 1)   GO TO   352                                   SPLIN157
      WRITE   (NTP6,60)   NSUP, ISP                                     SPLIN158
  352 WRITE   (NTP6,70)   (INSUP(I),   I=1,NSUP)                        SPLIN159
      IF   (IF1 .EQ. 1)   GO TO   370                                   SPLIN160
      IPAN     =   INSUP(1)                                             SPLIN161
      NSRUN    =   0                                                    SPLIN162
      DO   360   IPN = 1, IPAN                                          SPLIN163
      NSRUN    =   NSRUN + NSARAY(IPN)                                  SPLIN164
  360 CONTINUE                                                          SPLIN165
      ABSCOS =   ABS(CG(NSRUN))                                         SPLIN166
      WRITE   (NTP6,80)                                                 SPLIN167
  370 CONTINUE                                                          SPLIN168
      IF   (IF1 .EQ. 1)   WRITE   (NTP6,90)                             SPLIN169
      DO   400   I = 1, NXQ                                             SPLIN170
      ID         =   ID + 1                                             SPLIN171
      MODE     =   MODE + 1                                             SPLIN172
      INOD     =   NODE(I)                                              SPLIN173
      XIQ(I) =   ELXI(INOD)                                             SPLIN174
      ETAQ(I) = 0.0                                                     SPLIN175
      IF   (IF1 .EQ. 1)   GO TO   390                                   SPLIN176
      IF   (ABSCOS .LE. 0.00001)   GO TO   380                          SPLIN177
      ETAQ(I)=   ELYI(INOD)                                             SPLIN178
      GO TO   390                                                       SPLIN179
  380 CONTINUE                                                          SPLIN180
      ETAQ(I)=   ELZI(INOD)                                             SPLIN181
  390 CONTINUE                                                          SPLIN182
      WRITE   (NTP6,160)   ISP, ID, MODE, NODE(I), XIQ(I), ETAQ(I)      SPLIN183
  400 CONTINUE                                                          SPLIN184
      IF   (NACELL .EQ. 0)   GO TO   410                                SPLIN185
C                                                                       SPLIN186
C         NEW NACELLE OPTION   --   MAY 1977                            SPLIN187
C                                                                       SPLIN188
      CALL          NACL(ISP, NTPSH, NBEOLD, NBCUM, X3L  , NXQ , XIQ ,  SPLIN189
     1              NSUP, INSUP, H , DHDX  , NMODE, KPRINT   , NTP6   ) SPLIN190
C                                                                       SPLIN191
      GO TO   490                                                       SPLIN192
  410 CONTINUE                                                          SPLIN193
      IF   (IMDONE.NE. 3)   NMODE = NMODE + NXQ                         SPLIN194
      I1       = 1                                                      SPLIN195
      I2       = 1                                                      SPLIN196
      ISORT    = 1                                                      SPLIN197
      CALL          SORT(IF1,IF2,NSUP,INSUP,I1,I2,NXQ,LMAX,X, XIC, X3L,YL, SPLIN198
     1              XP,YP,H ,DHDX,   ISP, NSBP,NBCUM,NBARAY,NTPH,NTPDH, SPLIN199
     2              NTPSH,KPRINT,NTP6,KCUM1,KCUM2,KCUM3,NTOT,ISORT,NODE, SPLIN200
     3           WORK(NLOC),WORK(NEXT1)   )                             SPLIN201
      IF   (KPRINT .NE. 2)   GO TO   430                                SPLIN202
      IF   (IF1. EQ .1)   GO TO   420                                   SPLIN203
      WRITE   (NTP6,120)   I2, ISP                                      SPLIN204
      WRITE   (NTP6,150)   (XP(I),   I=I1,I2)                           SPLIN205
```

```
      WRITE  (NTP6,130)   I2, ISP                                      SPLIN206
      WRITE  (NTP6,150)   (YP(I),   I=I1,I2)                           SPLIN207
      GO TO  430                                                       SPLIN208
  420 WRITE  (NTP5,140)   I2, ISP                                      SPLIN209
      WRITE  (NTP6,150)   (XP(I),   I=I1,I2)                           SPLIN210
  430 CONTINUE                                                         SPLIN211
      REWIND  NTP1                                                     SPLIN212
C                                                                      SPLIN213
C   ***   THE INPUT ARRAY  ETAQ  IS NOW DIVIDED BY THE  'MODIFIED      SPLIN214
C         COSINE'  OF THE DIHEDRAL ANGLE OF THE FIRST PANEL  IN THE    SPLIN215
C         CURRENT  SUPERPANEL  FOR WHICH THE  ETA-Q ARRAY IS USED      SPLIN216
C                                                                      SPLIN217
      IF   ((IF1+IF2) .GE. 1)   GO TO  440                             SPLIN218
      IQ     = INSUP(1)                                                SPLIN219
      COSGAM = CSG(IQ)                                                 SPLIN220
      GO TO  450                                                       SPLIN221
  440 COSGAM = 1.0                                                     SPLIN222
  450 CONTINUE                                                         SPLIN223
      DO   460   IXQ=1,NXQ                                             SPLIN224
      ETAQ(IXQ) = ETAQ(IXQ) / COSGAM                                   SPLIN225
  460 CONTINUE                                                         SPLIN226
      CALL          SPL1(IF1,NXQ,LMAX,NTP1,NTP6,            XIQ,ETAQ,  SPLIN227
     1                   WORK(NLOC), WORK(NEXT),  IXCON,  IYCON,  NMAX    )  SPLIN228
      CALL          SPL2(IF1,IF2,NXQ,LMAX,NTP2,NTP3,NTP6,  KPRINT,   NTPH,  SPLIN229
     1              NTPDH,I2,XP,YP,XIQ,ETAQ,H,DHDX,                    SPLIN230
     2              WORK(NLOC),WORK(NEXT),NMAX)                        SPLIN231
      ISORT  = 2                                                       SPLIN232
      REWIND  NTPH                                                     SPLIN233
      REWIND  NTPDH                                                    SPLIN234
      NTPSH   = NTP9                                                   SPLIN235
      IF   (IMDONE . EQ. 3)   NTPSH = NTPSH4                           SPLIN236
      CALL          SORT(IF1,IF2,NSUP,INSUP,I1,I2,NXQ,LMAX,X, XIC, X3L,YL,  SPLIN237
     1              XP,YP,H ,DHDX,  ISP, NSBP,NBCUM,NBARAY,NTPH,NTPDH,  SPLIN238
     2              NTPSH,KPRINT,NTP6,KCUM1,KCUM2,KCUM3,NTOT,ISORT,NODE,  SPLIN239
     3           WORK(NLOC),WORK(NEXT1)   )                            SPLIN240
      NTPSH   = NTP9                                                   SPLIN241
      GO TO   (490,470,480),   IMDONE                                  SPLIN242
  470 CONTINUE                                                         SPLIN243
      IMDONE = 3                                                       SPLIN244
      IF2    = 1                                                       SPLIN245
      GO TO  410                                                       SPLIN246
  480 CONTINUE                                                         SPLIN247
      IMDONE = 2                                                       SPLIN248
  490 CONTINUE                                                         SPLIN249
      IF   (IMDONE . GT . 1)   GO TO  520                              SPLIN250
C                                                                      SPLIN251
C   ***   COPY CONTENTS OF TAPE  NTPSH  (H  AND DHDX  MATRIX COLUMNS   SPLIN252
C         FOR ALL BODY ELEMENTS  AND ALL 'MODES' )  ONTO THE MASTER    SPLIN253
C         TAPES  MASTH  AND  MASTDH  RESPECTIVELY, EACH PRECEDED BY THE SPLIN254
C         TOTAL NUMBER OF MODES,  NMODE.                               SPLIN255
      REWIND  NTPSH                                                    SPLIN256
      NMODEB   =  NMODE                                                SPLIN257
```

```
      DO  500   II = 1, NMODE
      READ    (NTPSH )  NODEK, NTOT, ( H   (I),   I = 1, NTOT)
      IF   (IFNEWH .NE. 0)  NODEK = II
      WRITE   (MASTH )  NODEK
      WRITE   (MASTH )           NTOT, ( H   (I),   I = 1, NTOT)
      READ    (NTPSH )  NODEX, NTOT, (DHDX(I),   I = 1, NTOT)
      WRITE   (MASTDH)  NODEK
      WRITE   (MASTDH)           NTOT, (DHDX(I),   I = 1, NTOT)
  500 CONTINUE
      WRITE   (MASTH )  KODE
      WRITE   (MASTDH)  KODE
      REWIND  NTPSH
      REWIND  NTPSH4
      NMODE   = 0
  510 CONTINUE
      IF   (NSP .EQ. 0 .OR. IFNEWH .NE. 0)   GO TO  540
      NSBP    = NSP
      NTOT    = NTP
      IMDONE = 2
      GO TO  300
  520 CONTINUE
C
C  ***    COPY CONTENTS OF TAPE  NTPSH  (H  AND DHDX MATRIX COLUMNS
C         FOR ALL LIFTING SURFACE BOXES  AND ALL 'MODES') ONTO THE
C         MASTER TAPES  MASTH  AND  MASTDH  RESPECTIVELY  PRECEDED
C         BY THE NUMBER OF MODES  NMODE  --  ALSO
C         COPY  H-1/4 COLUMNS FROM NTPSH4  ONTO THE MASTER TAPE
C         MASTH4,  PRECEDED BY  NMODE.
C
      REWIND  NTPSH
      REWIND  NTPSH4
C
      DO  530   II = 1, NMODE
      READ    (NTPSH )  NODEK, NTOT, ( H   (I),   I = 1, NTOT)
      WRITE   (MASTH )  NODEK
      WRITE   (MASTH )           NTOT, ( H   (I),   I = 1, NTOT)
      READ    (NTPSH )  NODEK, NTOT, (DHDX(I),   I = 1, NTOT)
      WRITE   (MASTDH)  NODEK
      WRITE   (MASTDH)           NTOT, (DHDX(I),   I = 1, NTOT)
      READ    (NTPSH4)  NODEK, NTOT, ( H   (I),   I = 1, NTOT)
      WRITE   (MASTH4)  NODEK
      WRITE   (MASTH4)           NTOT, ( H   (I),   I = 1, NTOT)
  530 CONTINUE
      WRITE   (MASTH )  KODE
      WRITE   (MASTDH)  KODE
      WRITE   (MASTH4)  KODE
  540 CONTINUE
C
      RETURN
      END
```

```fortran
      SUBROUTINE SPLN(SPLINF, SCOFF , THB   , XI    , ETA   ,
     1               XA, YA, XB, YB,       NODES , NDIM , IPRINT, NTP6      )
C
C         THIS SUBROUTINE COMPUTES THE *SPLINE* MATRIX,  AND THE *RHS*
C         MATRIX FOR A SUPERPANEL,  THEN CALL SUBROUTINE MIS1  TO SOLVE
C         FOR THE SPLINE COEFFICIENTS  MATRIX  *SCOEF*
C
      DIMENSION  SPLINE(NDIM, NDIM), SCOEF(NDIM, NODES), THB( 1 ),
     1               XI( 1 ), ETA( 1 )
C
      NODP1      =  NODES + 1
      NODP2      =  NODES + 2
      NODM1      =  NODES - 1
C
      IF  (NODES .EQ. 1)  GO TO  140
      DO  20  J = 1, 2
      DO  10  I = 1, 2
      SPLINE(I,J)  =  0.0
   10 CONTINUE
   20 CONTINUE
C
      DO  60  I = 1, NODP2
      IP2        =  I + 2
      SPLINE(IP2,1)  =  1.0
      SPLINE(1,IP2)  =  1.0
      IF  (I .GT. 1 .AND. I .LT. NODP2)  GO TO  40
      IF  (I .EQ. NODP2)  GO TO  30
      THB(I)     =  0.0
      GO TO  50
C
   30 CONTINUE
      THB(I)     =  SQRT((XB-XA)**2 + (YB-YA)**2)
      GO TO  50
   40 CONTINUE
      J          =  I - 1
      THB(I)     =  SQRT((XI(J)-XA)**2 + (ETA(J)-YA)**2)
   50 CONTINUE
      SPLINE(IP2,2)  =  THB(I)
      SPLINE(2,IP2)  =  THB(I)
   60 CONTINUE
C
      DO  80  J = 1, NODP1
      JP1        =  J + 1
      JP2        =  J + 2
      K          =  JP1
C
      DO  70  I = K, NODP2
      IP2        =  I + 2
      ARG        =  ABS(THB(I) - THB(J))
      AIJ = 0.0
      IF (ARG.LT..0001) GO TO 62
      ARG2       =  ARG**2
```

```
      AIJ          =   2.0*ARG2 * ALOG(ARG)
   62 CONTINUE
      SPLINE(IP2,JP2)  =   AIJ
      SPLINE(JP2,IP2)  =   AIJ
   70 CONTINUE
   80 CONTINUE
C
C          COMPUTE MATRIX OF RIGHT-HAND-SIDES  (SAVE IN ARRAY  *SCOEF* )
C
      B            =   -THB(2) / (THB(3)-THB(2))
      A            =   1.0 - B
C
      D            =   (THB(NODP2)-THB(NODES)) / (THB(NODP1)-THB(NODES))
      C            =   1.0 - D
C
      SCOEF(3, 1)   =   A
      SCOEF(3, 2)   =   B
      SCOEF(NDIM,NODM1)   =   C
      SCOEF(NDIM,NODES)   =   D
C
      DO  90  I = 1, NODES
      IP3          =   I + 3
      SCOEF(IP3,I)    =   1.0
C
   90 CONTINUE
C
      IF  (IPRINT .NE. 2)  GO TO  130
      LINES        =   45
      LIMIT        =   LINES
      LCUM         =   3
      WRITE   (NTP6,110)
      DO  100  J = 1, NDIM
      WRITE   (NTP6,120)  J, (SPLINE(I,J),   I = 1, NDIM)
      LCUM         =   LCUM + 5 + NDIM / 6
      IF  (LCUM .LT. LIMIT .OR. J .EQ. NDIM)  GO TO  100
      WRITE   (NTP6,110)
      LCUM         =   LCUM + 3
      LIMIT        =   LIMIT + LINES
  100 CONTINUE
C
  110 FORMAT  ( 1H1 /// 10X, 25H***   SPLINE MATRIX   ***   /          )
  120 FORMAT  ( / 10X, 6HCOLUMN , I3  //  ( 5X, 6E15.6 )              )
C
  130 CONTINUE
C
C          SOLVE FOR THE SPLINE COEFFICIENTS  *SCOEF*
C
      NMAX         =   NDIM
C
      CALL         MISI( SPLINE, NDIM, NMAX, SCOEF, NODES, NERR,  0      )
C
      IF  (IPRINT .EQ. 2)  GO TO  160
```

```
      RETURN                                                            SPLN 106
C                                                                       SPLN 107
  140 CONTINUE                                                          SPLN 108
      SCOEF(1,1)   =   1.0                                              SPLN 109
      DO  150  I = 2, 5                                                 SPLN 110
  150 SCOFF(I,1)   =   0.0                                              SPLN 111
C                                                                       SPLN 112
      RETURN                                                            SPLN 113
  160 CONTINUE                                                          SPLN 114
C                                                                       SPLN 115
      LINES     =   45                                                  SPLN 116
      LIMIT     =   LINES                                               SPLN 117
      LCUM      =   3                                                   SPLN 118
      WRITE   (NTP6,180)                                                SPLN 119
      DO  170  J = 1, NODES                                             SPLN 120
      WRITE   (NTP6,190)  J,  (SCOEF(I,J),  I = 1, NDIM)                SPLN 121
      LCUM      =   LCUM + 5 + NDIM / 6                                 SPLN 122
      IF  (LCUM .LT. LIMIT .OR. J .EQ. NODES)  GO TO  170               SPLN 123
      WRITE   (NTP6,180)                                                SPLN 124
      LIMIT     =   LIMIT + LINES                                       SPLN 125
      LCUM      =   LCUM + 3                                            SPLN 126
  170 CONTINUE                                                          SPLN 127
C                                                                       SPLN 128
  180 FORMAT  ( 1H1 /// 10X, 37H***   SPLINE COEFFICIENT MATRIX   ***/) SPLN 129
  190 FORMAT  ( / 10X, 6HCOLUMN , I3 //  ( 5X, 6E15.6 )               ) SPLN 130
C                                                                       SPLN 131
      RETURN                                                            SPLN 132
      END                                                               SPLN 133
```

```
      SUBROUTINE SPL1(IF1,NXQ,LMAX,NTP1,NTP6,            XIQ,ETAQ,
     *            RHS, XKD, IXCON, IYCON, NMAX                        )
      DIMENSION  XKD(NMAX, NMAX) ,  RHS (NMAX, NMAX)
      DIMENSION  XIQ( 1 ), ETAQ( 1 )
C
      IFLAG  = 0
      NUMBR  = 0
      NKD    = NXQ + 3
      BFLAG     = 0.0
      COSAV     = 0.0
      SINAV     = 0.0
      C2TIP1    = 0.0
      S2TIP1    = 0.0
      DO  70  IQ = 2, NXQ
      DELX2     = ( XIQ(IQ) -  XIQ(1))  ** 2
      DELY2     = (ETAQ(IQ) - ETAQ(1)) ** 2
      RHO       = DELX2 + DELY2
      C2TI      = DELX2 / RHO
      S2TI      = DELY2 / RHO
      IF  (IQ . EQ . 2)  GO TO  60
      EPS1      = ABS(C2TIP1 - C2TI)
      EPS2      = ABS(S2TIP1 -  S2TI)
      IF  (EPS1 . GE . 0.01 . OR . EPS2 . GE . 0.01)  GO TO  80
   60 C2TIP1    = C2TI
      S2TIP1    = S2TI
      COSAV     = COSAV + C2TI
      SINAV     = SINAV + S2TI
   70 CONTINUE
      SIGNX     = 1.0
      SIGNY     = 1.0
      XIDIF     =  XIQ(2) -  XIQ(1)
      ETDIF     = ETAQ(2) - ETAQ(1)
      IF  (XIDIF . EQ . 0.0)  GO TO  72
      SIGNX     = XIDIF / ABS(XIDIF)
   72 IF  (ETDIF . EQ . 0.0)  GO TO  75
      SIGNY     = ETDIF / ABS(ETDIF)
   75 CONTINUE
      CTH       = SIGNX *  SQRT(COSAV  / (NXQ - 1))
      STH       = SIGNY *  SQRT(SINAV  / (NXQ - 1))
      BFLAG     = 1.0
   80 CONTINUE
      DO  200  J=1,3
      XKD(1,J) = 0.0
      IF  (IF1 . NE . 0)  GO TO  120
      IF  (J . EQ . 1 . OR . BFLAG . EQ . 0.0)  GO TO  105
      IF  (J . EQ . 3)  GO TO  90
      XKD(2,J) =  STH
      XKD(3,J) = -STH
      GO TO  150
   90 CONTINUE
      XKD(2,J) = -CTH
      XKD(3,J) =  CTH
```

```
      GO TO  150
  105 XKD(2,J) = 0.0
  110 XKD(3,J) = 0.0
      GO TO  150
  120 IF  (IXCON . EQ . 1  . AND . J . EQ . 2)  GO TO  130
      IF  (IYCON . EQ . 1  . AND . J . EQ .3)  GO TO  140
      GO TO  105
  130 XKD(2,J) = 1.0
      GO TO  110
  140 XKD(2,J) = 0.0
      XKD(3,J) = 1.0
  150 CONTINUE
      DO  190  I=4,NKD
      IM3     = I - 3
      GO TO  (160, 170, 180), J
  160 XKD(I,J) = 1.0
      GO TO  190
  170 XKD(I,J) = XIQ( IM3)
      GO TO  190
  180 XKD(I,J) = ETAQ(IM3)
  190 CONTINUE
C   ***      WRITE  ONE COLUMN OF THE KD- MATRIX  (OUT OF THE FIRST 3)
      WRITE  (NTP1)      (XKD(I,J),  I=1,NKD)
  200 CONTINUE
      DO  300  J=4,NKD
      JM3     = J - 3
      XKD(1,J) = 1.0
      XKD(2,J) = XIQ( JM3)
      XKD(3,J) = ETAQ(JM3)
      DO  260  I=4,NKD
      IM3     = I - 3
      IF  (I . EQ . J)  GO TO  250
      IF  (XIQ(IM3). EQ .XIQ(JM3))  GO TO  220
      TRM1    = (XIQ(IM3)-XIQ(JM3))**2
  210 IF  (ETAQ(IM3).EQ.ETAQ(JM3))  GO TO  230
      TRM2    = (ETAQ(IM3)-ETAQ(JM3))**2
      GO TO  240
  220 TRM1    = 0.0
      GO TO  210
  230 TRM2    = 0.0
  240 RAIJ2   = TRM1 + TRM2
      AIJ     = RAIJ2 * ALOG(RAIJ2)
      GO TO  260
  250 AIJ     = 0.0
  260 XKD(I,J) = AIJ
C   ***      WRITE A COLUMN OF THE  KD-MATRIX
      WRITE  (NTP1)      (XKD(I,J),  I=1,NKD)
  300 CONTINUE
C
C   ***     KD MATRIX FORMATION FOR ONE SUPERPANEL (BODY)  COMPLETE
C
C   ***    FORM   RHS   OF EQ.  XXX
```

```
C
      DO  700  J=1,NXQ
      DO  650  I=1,NKD
      IM3    =  I - 3
      RHS(I,J) = 0.0
      IF  (IM3 . EQ . J)  RHS(I,J) = 1.0
  650 CONTINUE
  660 CONTINUE
  700 CONTINUF
C
C    ***   USE THE STANDARD  SUBROUTINE   MIS1   TO SOLVE EQ.  XXX
C
      NERR    = 0
      SCALR   = 0.0
      NKMAX   = NMAX
      CALL        MIS1(XKD, NKD, NKMAX, RHS, NXQ, NERR, SCALR)
C
C
      RETURN
      END
```

```
      SUBROUTINE SPL2(IF1,IF2,NXQ,LMAX,NTP2,NTP3,NTP6, KPRINT,  NTPH,
     1           NTPDH,I2,XP,YP,XIQ,ETAQ,XKF,DKFDX,CMAT,SUM,NMAX)
      DIMENSION  XP(LMAX), YP(LMAX), XIQ( 1 ),  ETAQ( 1 )
      DIMENSION XKF(1), DKFDX(1)
      DIMENSION  SUM( I2 , NMAX) ,  CMAT(NMAX, NMAX)
   25 FORMAT   ( 6E16.6 )
   35 FORMAT   ( 1H1 /// 6H   THE, I4, 3H BY, I4, 24H  H (3/4-CHORD)  MAT
     1RIX  /   )
   45 FORMAT   ( 1H1 /// 6H   THE, I4, 3H BY, I4, 24H  H (1/4-CHORD)  MAT
     1RIX  /   )
   55 FORMAT   ( 1H1 /// 6H   THE, I4, 3H BY, I4, 24H    DHDX         MAT
     1RIX  /   )
   60 FORMAT   ( /  9H   COLUMN , I3  /  )
      TRM1X   = 0.0
      DMUDX   = 0.0
      NKF     = NXQ + 3
C
      REWIND  NTP2
      REWIND  NTP3
      REWIND  NTPH
      REWIND  NTPDH
C
      NN      = I2
      DO  350  J=1,3
C     WRITE  (NTP6,35)  J, NN
      DO  340  I=1,NN
      GO TO  (310, 320, 330),  J
  310 XKF(I) = 1.0
      IF  (IF2 . NE . 0)  GO TO  340
      DKFDX(I) = 0.0
      GO TO  340
  320 XKF(I) = XP(I)
      IF  (IF2. EQ .0)  DKFDX(I) = 1.0
      GO TO  340
  330 XKF(I) = YP(I)
      IF  (IF2. EQ .0)  DKFDX(I) = 0.0
  340 CONTINUE
      WRITE  (NTP2)     (XKF(I),    I=1,NN)
      IF  (IF2. NE . 0)  GO TO  350
      WRITE  (NTP3)     (DKFDX(I), I=1,NN)
  350 CONTINUE
      DO  500  J=1,NXQ
      DO  400  I=1,NN
      IF  (XP(I). EQ . XIQ(J))  GO TO  370
      TRM1   = (XP(I) - XIQ(J))**2
      IF  (IF2. EQ .0)  TRM1X = XP(I)  - XIQ(J)
  360 IF  (YP(I). EQ . ETAQ(J))  GO TO  380
      TRM2   = (YP(I) -ETAQ(J))**2
      GO TO  385
  370 TRM1   = 0.0
      TRM1X  = 0.0
      GO TO  360
```

```
  380 TRM2    = 0.0
  385 RBIJ2   = TRM1 + TRM2
      IF  (RBIJ2. LT . 0.0001)  GO TO  387
      XMUIJ  = RBIJ2 * ALOG(RBIJ2)
      IF  (IF2. NE . 0)  GO TO  389
      IF  (IF1. EQ . 0)  DMUDX = 2.0 * TRM1X * ALOG(RBIJ2)
      IF  (IF1. EQ . 1)  DMUDX = 2.0 * TRM1X * ALOG(TRM1)
      GO TO  390
  387 XMUIJ  = 0.0
  389 DMUDX  = 0.0
  390 CONTINUE
      XKF(I) = XMUIJ
      DKFDX(I) = DMUDX
  400 CONTINUE
      WRITE  (NTP2)       (XKF(I),    I=1,NN)
      IF  (IF2. NE . 0)  GO TO  500
      WRITE  (NTP3)       (DKFDX(I), I=1,NN)
  500 CONTINUE
C
      DO  620  J=1,NXQ
      DO  610  I=1,NN
  610 SUM(I,J) = 0.0
  620 CONTINUE
C
      REWIND  NTP2
      DO  650  J=1,NKF
      READ   (NTP2)   (XKF(II),   II=1,NN)
      DO  640  I=1,NN
      DO  630  JC=1,NXQ
      SUM(I,JC) = SUM(I,JC) + XKF(I) * CMAT(J,JC)
  630 CONTINUE
  640 CONTINUF
  650 CONTINUE
C
C   ***   COMPUTATION OF THE  -H-  MATRIX COMPLETF
C
C   ***   WRITE  MATRIX COLUMNS ON  TAPE   NTPH
C
      IF  (KPRINT .EQ. 0)  GO TO  660
      IF  (IF2 .EQ. 0)  WRITE  (NTP6,35)  NN, NXQ
      IF  (IF2 .NE. 0)  WRITE  (NTP6,45)  NN, NXQ
  660 CONTINUE
      DO  670  J=1,NXQ
      WRITE  (NTPH)       (SUM(I,J),  I=1,NN)
      IF  (KPRINT .EQ. 0)  GO TO  670
      WRITE  (NTP6,60)  J
      WRITE  (NTP6,25)  (SUM(I,J),  I=1,NN)
  670 CONTINUE
C
C   ***
C
      IF  (IF2 . NE . 0)  GO TO  800
```

```
      REWIND  NTP3
      DO   720  J=1,NXQ
      DO   710  I=1,NN
  710 SUM(I,J) = 0.0
  720 CONTINUE
      DO   750  J=1,NKF
      READ    (NTP3)    (DKFDX(II),   II=1,NN)
      DO   740  I=1,NN
      DO   730  JC=1,NXQ
      SUM(I,JC) = SUM(I,JC) + DKFDX(I) * CMAT(J,JC)
  730 CONTINUE
  740 CONTINUE
  750 CONTINUE
C
C   ***    COMPUTATION OF THE  -DHDX-   MATRIX COMPLETE
C
C   ***    WRITE  MATRIX COLUMNS ON  TAPE   NTPDH
C
      IF  (KPRINT .NE. 0)  WRITE  (NTP6,55)  NN, NXQ
      DO   770  J=1,NXQ
      WRITE  (NTPDH)     (SUM(I,J),   I=1,NN)
      IF  (KPRINT .EQ. 0)  GO TO  770
      WRITE  (NTP6,60)  J
      WRITE  (NTP6,25)   (SUM(I,J),   I=1,NN)
  770 CONTINUE
C
  800 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SUBB(KB,KS,I,J,JZ,JB,LB,LS,NDY,NYFL,FLND,FLNE,PI,EPS,    SUBB    2
     1      SGR,CGR,SGS,CGS,AR,SL,CL,TL,FL,BETA,SUM                       SUBB    3
     2 , YB, ZB, RIA, X, YS, ZS, DELX )                                   SUBB    4
      DIMENSION YB(1), ZB(1), RIA(1), X(1), YS(1), ZS(1), DELX(1)         SUBB    5
C    ***    COMPUTES ELEMENTS OF THE SUBMATRICES  DZP, DZZ, DZY, DYP,     SUBB    6
C           DYZ  AND  DYY  USING  SUBROUTINE  DZY                         SUBB    7
      COMMON /AROCOM/ NTI, MODES                                          AROCOM  2
     X , NP, MSTRIP, NSMAX, NCMAX, NTOTAL, NB, MSBE, MBE                  AROCOM  3
     Y , ND, NE, NBY, NBZ, NTO, NTP, NTY, NTZ                             AROCOM  4
     1 , NTYS, NTZS, MAXGR, MAXSTR, NSBETO, NSTRIP, KR, XM, REFA, REFC    AROCOM  5
     2 , REFS, FMACH, LINES                                               AROCOM  6
      COMMON      /KCS/ IND,KD1R,KD1I,KD2R,KD2I                           SUBB    9
      REAL        KD1R,KD1I, KD2R,KD2I                                    SUBB   10
      REAL        KR,M                                                    SUBB   11
      COMPLEX     DPUR,DPUL,DPLR,DPLL,DP,SUM(2),EIKX                      SUBB   12
   10 FORMAT  (1H0,6E20.8)                                                SUBB   13
      IPRNT  = 0                                                          SUBB   14
      IND  = 0                                                            SUBB   15
      BR   = FL/2.0                                                       SUBB   16
      DPUR = (0.0,0.0)                                                    SUBB   17
      DPUL = (0.0,0.0)                                                    SUBB   18
      DPLR = (0.0,0.0)                                                    SUBB   19
      DPLL = (0.0,0.0)                                                    SUBB   20
      ANOT = RIA(JB)                                                      SUBB   21
      DXS  = DELX(J)                                                      SUBB   22
      ABSYB= ABS(YB(LB))                                                  SUBB   23
      ABSZB= ABS(ZB(LB))                                                  SUBB   24
                                                                          SUBB   25
                                                                          SUBB   26
      IF (KB.EQ.0)  GO TO  20                                             SUBB   27
      TEST1= ABS(YB(LB) -YB(KB))                                          SUBB   28
      TEST2= ABS(ZB(LB) -ZB(KB))                                          SUBB   29
      IF  (TEST1.GT.EPS. OR .TEST2.GT.EPS)  GO TO  20                     SUBB   30
      SUM(1) = (0.0,0.0)                                                  SUBB   31
      SUM(2) = (0.0,0.0)                                                  SUBB   32
      IF (NDY.NE.NYFL)  GO TO  100                                        SUBB   33
      IF (I.NE.J)   GO TO 100                                             SUBB   34
      D2D  =        1.0 /(2.0*PI*ANOT*ANOT*(1.0+AR))                      SUBB   35
      IF     (NDY.NE.0)  D2D=D2D/AR                                       SUBB   36
      SUM(1)= CMPLX(D2D,0.0)                                              SUBB   37
      SUM(2)= CMPLX(D2D,0.0)                                              SUBB   38
      GO TO  100                                                          SUBB   39
   20 CONTINUE                                                            SUBB   40
      XX    = X(I)                                                        SUBB   41
      Y     = YS(KS)                                                      SUBB   42
      Z     = ZS(KS)                                                      SUBB   43
      XI1   = X(J) - 0.5*DXS                                              SUBB   44
      XI2   = X(J) + 0.5*DXS                                              SUBB   45
      ETA   = YS(LS)                                                      SUBB   46
      ZETA  = ZS(LS)                                                      SUBB   47
      AO    = ANOT                                                        SUBB   48
      IDZDY= NDY                                                          SUBB   49
```

```
      IGO   = 1
      LHS       =   0
   30 CONTINUE
      CALL          DZY  (XX, Y, Z, SGR, CGR, XI1, XI2, ETA, ZETA, AR, AO,
     1        KR    ,   REFC    ,    BETA   ,    FMACH ,
     2         IDZDY ,    DZDYR ,    DZDYI ,    LHS    ,    IPRNT )
      DP    = CMPLX(DZDYR,DZDYI)
      GO TO  (40,50,70,80),   IGO
   40 CONTINUE
C   UPPER RIGHT-HAND SIDE CONTRIBUTION
      DPUR = DP
      IF (KB.EQ.LB)  GO TO 100
      IF (ABSYB.LE.EPS.AND.ABSZB.LE.EPS)  GO TO  90
      IF             (ABSYB.LE.EPS)  GO TO  60
      IGO   = 2
      ETA   = -YS(LS)
      LHS       =  1
      GO TO  30
   50 CONTINUE
C   UPPER LEFT-HAND SIDE CONTRIBUTION
      DPUL = DP
      LHS       =  0
   60 CONTINUE
      IF (NE.EQ.0.OR.ABSZB.LE.EPS)  GO TO  90
      IGO   = 3
      ETA   =  YS(LS)
      ZETA = -ZS(LS)
      GO TO  30
   70 CONTINUE
C   LOWER RIGHT-HAND SIDE CONTRIBUTION
      DPLR = DP
      IF             (ABSYB.LE.EPS)  GO TO  90
      IGO   = 4
      ETA   = -YS(LS)
      ZETA = -ZS(LS)
      LHS       =  1
      GO TO  30
   80 CONTINUE
C   LOWER  LEFT-HAND SIDE CONTRIBUTION
      DPLL = DP
   90 CONTINUE
      SUM(1)=DPUR + DPUL + FLNE*DPLR + FLNE*DPLL
      SUM(2)=DPUR - DPUL + FLNE*DPLR - FLNE*DPLL
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SUBI(DA,DZB,DYB,DAR,DETA,DZETA,DCGAM,DSGAM,DEE,DXI,TL,  SUBI
     1 DETAI,DZETAI,DCGAMI,DSGAMI,DEEI,DTLAMI,DMUY,DMUZ,INFL,IOUTFL)     SUBI
C   ***    COMPUTES THE IMAGE POINT COORDINATES INSIDE ASSOCIATED        SUBI
C          BODIES AND THE  MU-Z  MU-Y  ELEMENTS USED IN SUBROUTINE FWMW  SUBI
   15 FORMAT  (1H0,15I5)                                                 SUBI
   35 FORMAT  (1H0,6E20.8)                                               SUBI
      EPS  = 0.1*DEE                                                     SUBI
      DMUY = 0.0                                                         SUBI
      DMUZ = 0.0                                                         SUBI
      IGO  = 1                                                           SUBI
      PSQR = SQRT(((DETA-DYB)*DAR)**2 + (DZETA-DZB)**2)                  SUBI
      COSTH= (DETA -DYB)*DAR/PSQR                                        SUBI
      SINTH= (DZETA-DZB)/PSQR                                            SUBI
      CT2  = COSTH*COSTH                                                 SUBI
      ST2  = SINTH*SINTH                                                 SUBI
      CT3  = COSTH*CT2                                                   SUBI
      ST3  = SINTH*ST2                                                   SUBI
      YCBAR= DA*(1.0-DAR*DAR)*CT3 + DYB                                  SUBI
      ZCBAR= DA*(DAR*DAR-1.0)*ST3/DAR + DZB                              SUBI
      PAREN= ST2 + DAR*DAR*CT2                                           SUBI
      PAR3 = PAREN*PAREN**2                                              SUBI
      ABAR = DA*SQRT(PAR3)/DAR                                           SUBI
      ABAR2= ABAR*ABAR                                                   SUBI
      IF (INFL.NE.0)  GO TO  300                                         SUBI
      ETA1 = DETA - DEE*DCGAM                                            SUBI
      ETA2 = DETA + DEE*DCGAM                                            SUBI
      ZETA1= DZETA- DEE*DSGAM                                            SUBI
      ZETA2= DZETA+ DEE*DSGAM                                            SUBI
      RHO12= (ETA1 - YCBAR)**2 + (ZETA1-ZCBAR)**2                        SUBI
      RHO22= (ETA2 - YCBAR)**2 + (ZETA2-ZCBAR)**2                        SUBI
      ETAI1= YCBAR + (ETA1-YCBAR)*ABAR2/RHO12                            SUBI
      ETAI2= YCBAR + (ETA2-YCBAR)*ABAR2/RHO22                            SUBI
      ZETI1= ZCBAR +(ZETA1-ZCBAR)*ABAR2/RHO12                            SUBI
      ZETI2= ZCBAR +(ZETA2-ZCBAR)*ABAR2/RHO22                            SUBI
      DEEI = SQRT((ETAI2-ETAI1)**2 + (ZETI2-ZETI1)**2) / 2.0             SUBI
      DETAI  = (ETAI1 + ETAI2)/2.0                                       SUBI
      DZETAI = (ZETI1 + ZETI2)/2.0                                       SUBI
      DCGAMI = -(ETAI2 - ETAI1)/(2.0*DEEI)                               SUBI
      DSGAMI= -(ZETI2 - ZETI1)/(2.0*DEEI)                                SUBI
      DXI1 = DXI - DEE*TL                                                SUBI
      DXI2 = DXI + DEE*TL                                                SUBI
      DELXI  = DXI1 - DXI2                                               SUBI
      DTLAMI = DELXI/(2.0*DEEI)                                          SUBI
      IF ((ABS(DAR-1.0)).LE.0.0001)  GO TO  420                          SUBI
      GO TO  301                                                         SUBI
  300 CONTINUE                                                           SUBI
      RHO2 = (DETA -YCBAR)**2 + (DZETA -ZCBAR)**2                        SUBI
      RHO4 = RHO2*RHO2                                                   SUBI
      DETAI  = YCBAR + (DETA -YCBAR)*ABAR2/RHO2                          SUBI
      DZETAI = ZCBAR + (DZETA-ZCBAR)*ABAR2/RHO2                          SUBI
  301 CONTINUE                                                           SUBI
      GO TO  (302,303,304),  IGO                                         SUBI
```

```
302 CONTINUE
    XETAI  = DETAI
    XZETAI = DZETAI
    GO TO  307
303 CONTINUE
    XETAI  = ETAI1
    XZETAI = ZETI1
    GO TO  307
304 CONTINUE
    XETAI  = ETAI2
    XZETAI = ZETI2
307 CONTINUE
    IF (DAR.LT.1.0)  GO TO  310
    DYBM = DYB-EPS
    DYBP = DYB+EPS
    IF  (DETA .GE.DYB .AND.XETAI .LT.DYBM)  GO TO  325
    IF  (DETA .LE.DYB .AND.XETAI .GT.DYBP)  GO TO  325
    GO TO  320
310 CONTINUE
    DZBM = DZB-EPS
    DZBP = DZB+EPS
    IF  (DZETA.GE.DZB .AND.XZETAI.LT.DZBM)  GO TO  325
    IF  (DZETA.LE.DZB .AND.XZETAI.GT.DZBP)  GO TO  325
320 CONTINUE
    PART1 = ((XETAI  - DYB) / DA)**2
    PART2 = ((XZETAI - DZB) / (DA*DAR))**2
    TEDIF = PART1 + PART2 - 1.0
    IF (INFL.EQ.0)  GO TO  400
    IF (TEDIF.LE.EPS)  GO TO  330
325 CONTINUE
    IOUTFL = 0
    GO TO  500
330 CONTINUE
    IOUTFL = 1
    TRM1 = (DETA -YCBAR)**2 - (DZETA -ZCBAR)**2
    TRM2 = 2.0 *(DETA-YCBAR)*(DZETA-ZCBAR)
    DMUY = -(-DSGAM*TRM1 + DCGAM*TRM2)*ABAR2/RHO4
    DMUZ = -(-DSGAM*TRM2 - DCGAM*TRM1)*ABAR2/RHO4

    GO TO 500
400 CONTINUE
    IF (TEDIF.GT.EPS)  GO TO  325
    IF  (IGO.EQ.3)  GO TO 420
    IGO  = IGO+1
    GO TO  301
420 CONTINUE
    IOUTFL = 1
500 CONTINUE
    RETURN
    END
```

```
      SUBROUTINE SUBP (I,L,LS,J,IO,IR,NBXS,NCPNB,SGR,CGR,YREC,ZREC,SUM  SUBP    2
     1 , NCARAY,YB,ZB,ARB,AVR,XLE,XTE,X,CG,EE,SG,YS,ZS,XIC,DELX,XLAM    SUBP    3
     2 , SCALER                                                        ) SUBP    4
      DIMENSION NCARAY(1), YB(1), ZB(1), ARB(1), AVR(1), XLE(1), XTE(1) SUBP    5
      DIMENSION X(1), CG(1), EE(1), SG(1), YS(1), ZS(1), XIC(1), DELX(1)SUBP    6
      DIMENSION XLAM(1)                                                 SUBP    7
C  ***    COMPUTES ELEMENTS OF THE SUBMATRICES  DPP, DPZ  AND  DPY      SUBP    8
C         USING  SUBROUTINES  SNPDF,  INCRO  AND SUBI                   SUBP    9
      COMMON /AROCOM/ NTI, MODES                                        AROCOM 2
     X , NP, MSTRIP, NSMAX, NCMAX, NTOTAL, NB, MSBE, MBE                AROCOM 3
     Y , ND, NE, NBY, NBZ, NTO, NTP, NTY, NTZ                           AROCOM 4
     1 , NTYS, NTZS, MAXGR, MAXSTR, NSBETO, NSTRIP, KR, XM, REFA, REFC  AROCOM 5
     2 , REFS, FMACH, LINES                                             AROCOM 6
      COMPLEX     DPUR,DPUL,DPLR,DPLL,DP,SUM(2)                         SUBP   11
      COMMON NAA,WORK(1)                                                SUBP   12
      REAL        KR,M                                                  SUBP   13
C                                                                       SUBP   14
   10 FORMAT  (1HO,15I5)                                                SUBP   15
   20 FORMAT  (1HO,6E20.8)                                              SUBP   16
C                                                                       SUBP   17
      NWK1 = NAA                                                        SUBP   18
      NWK2 = NWK1  + 14*NCMAX                                           SUBP   19
      NWK3 = NWK2  + 14*NCMAX                                           SUBP   20
      NWK4 = NWK3  + 14*NCMAX                                           SUBP   21
      NWK5 = NWK4  + 14*NCMAX                                           SUBP   22
      NWK6 = NWK5  + 14*NCMAX                                           SUBP   23
      NWK7 = NWK6  + 14*NCMAX                                           SUBP   24
      NWK8 = NWK7  + 14*NCMAX                                           SUBP   25
      NTOT      =  NTP + NTZ + NTY                                      SUBP   26
      PI   = 3.1415926                                                  SUBP   27
      EPS  = 0.00001                                                    SUBP   28
      M    = FMACH                                                      SUBP   29
      BETA = SQRT(1.0-M*M)                                              SUBP   30
      FL   = REFC                                                       SUBP   31
      NBV  = 0                                                          SUBP   32
      FLND = FLOAT(ND)                                                  SUBP   33
      FLNE = FLOAT(NE)                                                  SUBP   34
      SGS  = SG(LS)                                                     SUBP   35
      CGS  = CG(LS)                                                     SUBP   36
      DPUR = (0.0,0.0)                                                  SUBP   37
      DPUL = (0.0,0.0)                                                  SUBP   38
      DPLR = (0.0,0.0)                                                  SUBP   39
      DPLL = (0.0,0.0)                                                  SUBP   40
      DIJ  = 0.0                                                        SUBP   41
      DELR = 0.0                                                        SUBP   42
      DELI = 0.0                                                        SUBP   43
      DIJI = 0.0                                                        SUBP   44
      DELRI= 0.0                                                        SUBP   45
      DELII= 0.0                                                        SUBP   46
      INFL    = 0                                                       SUBP   47
      IOUTFL  = 0                                                       SUBP   48
      IMGS    = 0                                                       SUBP   49
```

```
C     UPPER RIGHT SENDING POINT                                         SUBP  50
      IGO   = 1                                                         SUBP  51
      LHS   = 0                                                         SUBP  52
      NDBLF= 0                                                          SUBP  53
      TL    = XLAM(J)                                                   SUBP  54
      SQTL  = SQRT(1.0+TL**2)                                           SUBP  55
      SL    = TL / SQTL                                                 SUBP  56
      CL    = 1.0/ SQTL                                                 SUBP  57
      XO    = X(I)   - XIC(J)                                           SUBP  58
      YO    = YREC -YS(LS)                                              SUBP  59
      ZO    = ZREC -ZS(LS)                                              SUBP  60
      ES    = EE(LS)                                                    SUBP  61
      DXS   = DELX(J)                                                   SUBP  62
      AX    = XO                                                        SUBP  63
      AY    = YO                                                        SUBP  64
      AZ    = ZO                                                        SUBP  65
      CV    = DXS                                                       SUBP  66
   30 CONTINUE                                                          SUBP  67
      IMG   = 0                                                         SUBP  68
      NOBI  = 1                                                         SUBP  69
      NA2   = 0                                                         SUBP  70
      CALL        SNPDF(SL,CL,TL,SGS,CGS,SGR,CGR,XO,YO,ZO,ES,DIJ,BETA,CV)SUBP  71
      IF (KR.LE.EPS)  GO TO  40                                         SUBP  72
      SDELX= DXS                                                        SUBP  73
      DELY = 2.0*ES                                                     SUBP  74
      AX1   = AX  + ES*TL                                               SUBP  75
      AY1   = AY  + ES*CGS                                              SUBP  76
      AZ1   = AZ  + ES*SGS                                              SUBP  77
      AX2   = AX  - ES*TL                                               SUBP  78
      AY2   = AY  - ES*CGS                                              SUBP  79
      AZ2   = AZ  - ES*SGS                                              SUBP  80
      CALL        INCRO(AX,AY,AZ,AX1,AY1,AZ1,AX2,AY2,AZ2,SGR,CGR,SGS,CGS,SUBP  81
     1        KR,FL,BETA,SDELX,DELY,DELR,DELI,IO,IR,NBXS,NCPNB,LHS,     SUBP  82
     2        NDBLE,IMG,NOBI,IMGS,  WORK(NWK1),WORK(NWK2),WORK(NWK3),   SUBP  83
     3        WORK(NWK4),WORK(NWK5),WORK(NWK6),WORK(NWK7),WORK(NWK8))   SUBP  84
   40 CONTINUE                                                          SUBP  85
      IF  (NB.EQ.0)  GO TO  12C                                         SUBP  86
C   **  CHECK FOR ASSOCIATED BODIES                                     SUBP  87
      DIJS = DIJ                                                        SUBP  88
      DELRS= DELR                                                       SUBP  89
      DELIS= DELI                                                       SUBP  90
      DIJI = 0.0                                                        SUBP  91
      DELRI= 0.0                                                        SUBP  92
      DELII= 0.0                                                        SUBP  93
      NA1   = 1                                                         SUBP  94
      NA2   = NB                                                        SUBP  95
      IMG   = 1                                                         SUBP  96
C   **  START DO-LOOP FOR THE SUMMATION OF THE WING-IMAGE CONTRIBUTIONS SUBP  97
C     OVER  RANGE(P)                                                    SUBP  98
      DO 110  NA=NA1,NA2                                                SUBP  99
      NOB   =  NA                                                       SUBP 100
C   **  NOB IS THE SEQUENCE NUMBER OF THE CURRENT BODY ASSOCIATED WITH  SUBP 101
```

```
C   PANEL  L   IN WHICH THE SENDING POINT  J  LIES                      SUBP 102
      NOBI = NOB                                                        SUBP 103
      DA   = AVR(NOB)                                                   SUBP 104
      DAR  = ARB(NOB)                                                   SUBP 105
      DXLE = XLE(NOB)                                                   SUBP 106
      DXTE = XTE(NOB)                                                   SUBP 107
      GO TO  (50,60,70,80),   IGO                                       SUBP 108
   50 CONTINUE                                                          SUBP 109
      DZB  = ZB(NOB)                                                    SUBP 110
      DYB  = YB(NOB)                                                    SUBP 111
      DETA = YS(LS)                                                     SUBP 112
      DZETA= ZS(LS)                                                     SUBP 113
      GO TO  90                                                         SUBP 114
   60 CONTINUE                                                          SUBP 115
      DZB  = ZB(NOB)                                                    SUBP 116
      DYB  =-YB(NOB)                                                    SUBP 117
      DETA =-YS(LS)                                                     SUBP 118
      DZETA= ZS(LS)                                                     SUBP 119
      GO TO  90                                                         SUBP 120
   70 CONTINUE                                                          SUBP 121
      DZB  =-ZB(NOB)                                                    SUBP 122
      DYB  = YB(NOB)                                                    SUBP 123
      DETA = YS(LS)                                                     SUBP 124
      DZETA=-ZS(LS)                                                     SUBP 125
      GO TO  90                                                         SUBP 126
   80 CONTINUE                                                          SUBP 127
      DZB  =-ZB(NOB)                                                    SUBP 128
      DYB  =-YB(NOB)                                                    SUBP 129
      DETA =-YS(LS)                                                     SUBP 130
      DZETA=-ZS(LS)                                                     SUBP 131
   90 CONTINUE                                                          SUBP 132
      DCGAM= CGS                                                        SUBP 133
      DSGAM= SGS                                                        SUBP 134
      DEE  = ES                                                         SUBP 135
      DXI  = XIC(J)                                                     SUBP 136
      IF  (DXI.LT.DXLE.OR.DXI.GT.DXTE)  GO TO 110                       SUBP 137
      TEST = SQRT((DETA-DYB)**2 + (DZETA-DZB)**2)                       SUBP 138
      IF  (TEST .GT. SCALER  *  DA)  GO TO  110                         SUBP 139
      CALL        SUBI(DA,DZB,DYB,DAR,DETA,DZETA,DCGAM,DSGAM,DEE,DXI,TL, SUBP 140
     1 DETAI,DZETAI,DCGAMI,DSGAMI,DEEI,DTLAMI,DMUY,DMUZ,INFL,IOUTFL)    SUBP 141
      DIJ  = 0.0                                                        SUBP 142
      IF  (INFL.NE.0.OR.IOUTFL.EQ.0)  GO TO 100                         SUBP 143
      DTL     = DTLAMI                                                  SUBP 144
      DSQRTL  = SQRT(1.0+DTL**2)                                        SUBP 145
      DSL     = DTL/DSQRTL                                              SUBP 146
      DCL     = 1.0/DSQRTL                                              SUBP 147
      XOI   = XO                                                        SUBP 148
      YOI   = YREC - DETAI                                              SUBP 149
      ZOI   = ZREC - DZETAI                                             SUBP 150
      CALL        SNPDF(DSL,DCL,DTL,DSGAMI,DCGAMI,SGR,CGR,XOI,YOI,ZOI,  SUBP 151
     1     DEEI,DIJ,BETA,CV)                                            SUBP 152
      DIJI = DIJI+DIJ                                                   SUBP 153
```

```
      IF  (KR.LE.EPS)  GO TO 100                                        SUBP 154
      DELR = 0.0                                                        SUBP 155
      DELI = 0.0                                                        SUBP 156
      AYI  = YOI                                                        SUBP 157
      AZI  = ZOI                                                        SUBP 158
      AY1I = AYI  -  DEEI*DCGAMI                                        SUBP 159
      AZ1I = AZI  -  DEEI*DSGAMI                                        SUBP 160
      AY2I = AYI  +  DEEI*DCGAMI                                        SUBP 161
      AZ2I = AZI  +  DEEI*DSGAMI                                        SUBP 162
      DEEI2=  2.0*DEEI                                                  SUBP 163
      CALL        INCRO(AX,AYI,AZI,AX1,AY1I,AZ1I,AX2,AY2I,AZ2I,SGR,CGR, SUBP 164
     1    DSGAMI,DCGAMI,KR,FL,BETA,SDELX,DEEI2,DELR,DELI,IO,IR,NBXS,    SUBP 165
     2  NCPNB,LHS,NDBLE,IMG,NCBI,IMGS,WORK(NWK1),WORK(NWK2),WORK(NWK3), SUBP 166
     3    WORK(NWK4),WORK(NWK5),WORK(NWK6),WORK(NWK7),WORK(NWK8))       SUBP 167
      DELRI= DELRI+DELR                                                 SUBP 168
      DELII= DELII+DELI                                                 SUBP 169
      IF  (IO . EQ . NCARAY(L))  IMGS=1                                 SUBP 170
      GO TO  110                                                        SUBP 171
  100 CONTINUE                                                          SUBP 172
      DELRI= 0.0                                                        SUBP 173
      DELII= 0.0                                                        SUBP 174
  110 CONTINUE                                                          SUBP 175
      DIJ  = DIJS                                                       SUBP 176
      DELR = DELRS                                                      SUBP 177
      DELI = DELIS                                                      SUBP 178
  120 CONTINUE                                                          SUBP 179
      DP   = CMPLX(((DIJ +DIJI)-(DELR +DELRI)), (-DELI -DELII))         SUBP 180
  130 CONTINUE                                                          SUBP 181
      GO TO  (140,150,170,180),  IGO                                    SUBP 182
  140 CONTINUE                                                          SUBP 183
      DPUR = DP                                                         SUBP 184
      IF            (ABS(YS(LS)) .LE.0.001)  GO TO  160                 SUBP 185
C   UPPER LEFT  SENDING POINT                                           SUBP 186
      IGO   = 2                                                         SUBP 187
      LHS   = 1                                                         SUBP 188
      SGS   = -SGS                                                      SUBP 189
      TL    = -TL                                                       SUBP 190
      SL    = -SL                                                       SUBP 191
      YO    = YREC  + YS(LS)                                            SUBP 192
      AY    = YO                                                        SUBP 193
      GO TO   30                                                        SUBP 194
  150 CONTINUE                                                          SUBP 195
      DPUL = DP                                                         SUBP 196
  160 CONTINUE                                                          SUBP 197
      IF (NE.EQ.0.OR.(ABS(ZS(LS))).LE.0.001)  GO TO  190                SUBP 198
C   LOWER RIGHT SENDING POINT                                           SUBP 199
      IGO   = 3                                                         SUBP 200
      LHS   = 0                                                         SUBP 201
      NDBLE= 1                                                          SUBP 202
      TL    =  XLAM(J)                                                  SUBP 203
      SL    =  TL/(SQRT(1.0+TL*TL))                                     SUBP 204
      YO    = YREC -YS(LS)                                              SUBP 205
```

```
      ZO    = ZREC + ZS(LS)                                           SUBP 206
      AY    = YO                                                      SUBP 207
      AZ    = ZO                                                      SUBP 208
      SGS   =-SG(LS)                                                  SUBP 209
      GO TO   30                                                      SUBP 210
  170 CONTINUE                                                        SUBP 211
      DPLR = DP                                                       SUBP 212
      IF             (ABS(YS(LS)) .LE.0.001)   GO TO  190             SUBP 213
C  LOWER LEFT  SENDING POINT                                          SUBP 214
      IGO   = 4                                                       SUBP 215
      LHS   = 1                                                       SUBP 216
      SGS   = SG(LS)                                                  SUBP 217
      TL    = -XLAM(J)                                                SUBP 218
      SL    =  TL/(SQRT(1.0+TL*TL))                                   SUBP 219
      YO    = YREC + YS(LS)                                           SUBP 220
      AY    = YO                                                      SUBP 221
      GO TO   30                                                      SUBP 222
  180 CONTINUE                                                        SUBP 223
      DPLL = DP                                                       SUBP 224
  190 CONTINUE                                                        SUBP 225
      SUM(1)= DPUR + DPUL + FLNE*DPLR + FLNE*DPLL                     SUBP 226
      SUM(2)= DPUR - DPUL + FLNE*DPLR - FLNE*DPLL                     SUBP 227
      RETURN                                                          SUBP 228
      END                                                             SUBP 229
```

```
      SUBROUTINE TKER  (X0,Y0,Z0,KR,BR,SGR,CGR,SGS,CGS,T1,T2,M)
C  ***    COMPUTES EITHER THE TOTAL KERNELS  (IND=0)  USED IN THE
C         CALCULATION OF A FINITE LENGTH DOUBLET LINE,  OR THE
C         INCREMENTAL OSCILLATORY KERNELS  (IND=1)  USED IN EVALUATING
C         THE INFLUENCE COEFFICIENT MATRIX ELEMENTS
      REAL    M, KR, KKR, KKI, IOOR, IOOI, JOOR, JOOI, I1OR, I1OI, I2OR3,
     1        I2OI3, IOUR, IOUI, JOUR, JOUI, I1UR, I1UI, I2UR3, I2UI3,
     2        K1, MU1, MU, K2
      REAL    K10,K20,K1RT1,K1IT1,K2RT2P,K2IT2P,K1OT1,K2OT2P
      COMMON  /DUM/  K10,K20,K1RT1,K1IT1,K2RT2P,K2IT2P,K1OT1,K2OT2P,E2
      COMMON     /KOS/ IND,KD1R,KD1I,KD2R,KD2I
      REAL       KD1R,KD1I, KD2R,KD2I
   10 FORMAT  (1H0,6F15.6/)
      EPS  = 0.00001
      PI = 3.1415926
      PI2= PI*PI
      K10     = 0.0
      K20     = 0.0
      K1RT1   = 0.0
      K1IT1   = 0.0
      K2RT2P  = 0.0
      K2IT2P  = 0.0
      K1OT1   = 0.0
      K2OT2P  = 0.0
      R1= SQRT  (Y0*Y0+Z0*Z0)
      R1S = R1
      IF  (ABS(R1) . GT . EPS)  GO TO  200
  100 IF  ( X0 )                   110,120,120
  110 KKR=0.
      KKI=0.
      GO TO  905
  120 C1= KR*X0/BR
      T1 = CGR*CGS + SGR*SGS
      KKR= 2.*(COS(C1)-1.)*T1
      KKI=-2.*SIN(C1)*T1
      K10     = 2.0
      K1RT1   = 2.0*T1*COS(C1)
      K1IT1   =-2.0*T1*SIN(C1)
      K1OT1   = 2.0*T1
      GO TO  905
  200 C1 = CGR
      C2 = SGR
      C3 = CGS
      C4 = SGS
      T2P=(Z0*Z0*C1*C3+Y0*Y0*C2*C4-Z0*Y0*(C2*C3+C1*C4))
      T2    = (100.*T2P)/(BR*BR)
      IF   ( ABS(T2)-EPS )                  210,220,220
  210 ICHUZ=1
      T1 = CGR*CGS + SGR*SGS
      T2 = 0.0
      GO TO 300
  220 T1 = CGR*CGS + SGR*SGS
```

```
      IF  ( ABS(T1)-EPS )                       230,240,240
  230 ICHUZ=2
      T1=0.
      GO TO 300
  240 ICHUZ=3
  300 BETA2 = (1.-M*M)
      BIGR = SQRT (X0*X0+BETA2*R1*R1)
      K1= KR*R1/BR
      MU1= (M*BIGR-X0)/ (BETA2*R1)
      MU=ABS(MU1)
      K2=K1*K1
      IF  ( MU1 )                  310,320,330
  310 ICHUZ=ICHUZ+3
      GO TO 330
  320 ICHUZ=ICHUZ+6
C    (N*C)**2  FOR  N=1,11  AND C=.372 =
C
C      .138384         .553536        1.245456        2.214144
C     3.4596          4.981824        6.780816        8.856576
C    11.209104       13.8384         16.744464
C
C    (N*C)  FOR  N=1,12  AND  14,16,18,20,22   =
C
C       .744          1.116          1.488          1.86          2.232
C      2.604          2.976          3.348          3.72          4.092
C      4.464          5.208          5.952          6.696         7.44
C      8.184
C
C    A(N)  FORN(=1,11  =
C
C      .24186198        -2.7918027        24.991979         -111.59196
C   271.43549        -305.75288        -41.18363           545.98537
C  -644.78155         328.72755        -64.279511
  330 CONTINUE
      EXARG = -0.372*MU
C    **  THE FOLLOWING TEST ON THE SIZE OF THE ARGUMENT TO  EXP  IS
C         NEEDED TO AVOID UNDERFLOW IN  SUBPROGRAM  EXP
      IF  (EXARG.GE.(-174.0))  GO TO  335
      E  = 0.0
      GO TO  337
  335 E  = EXP(EXARG)
      IF  (E . LE . 1.0E-30)  E = 0.0
  337 CONTINUE

      C1 = .138384+K2
      C2 = .553536+K2
      C3 = 1.245456+K2
      C4  = 2.214144 +K2
      C5  = 3.4596   +K2
      C6  = 4.981824 +K2
      C7  = 6.780816 +K2
      C8  = 8.856576 +K2
```

```
      C9  = 11.209104+K2
      C10 = 13.8384   +K2
      C11 = 16.744464+K2

      R1  = .24186198 / C1
      R2  =-2.7918027 / C2
      R3  = 24.991079 / C3
      R4  =-111.59196 / C4
      R5  = 271.43549 / C5
      R6  =-305.75288 / C6
      R7  =-41.18363  / C7
      R8  = 545.98537 / C8
      R9  =-644.78155 / C9
      R10 = 328.72755 / C10
      R11 =-64.279511 / C11

      IF  ( ICHUZ .LT. 4 )                              GO TO 340
      I00R = .372*(R1 +2.*R2 + 3.*R3 + 4.*R4 + 5.*R5 + 6.*R6 + 7.*R7 +
     1          8.*R8 + 9.*R9 + 10.*R10 + 11.*R11)
      I00I = -K1*(R1+R2+R3+R4+R5+R6+R7+R8+R9+R10+R11)

  340 GO TO  (420,350,350,390,350,350,380,350,350),ICHUZ

  350 Q1  = R1/ C1
      Q2  = R2/ C2
      Q3  = R3/ C3
      Q4  = R4/ C4
      Q5  = R5/ C5
      Q6  = R6/ C6
      Q7  = R7/ C7
      Q8  = R8/ C8
      Q9  = R9/ C9
      Q10 = R10/C10
      Q11 = R11/C11

      GO TO  (420,410,410,390,360,360,380,360,360),ICHUZ

  360 J00R =  Q1*(.138384-K2)+Q2*(.553536-K2)+Q3*(1.245456-K2)+Q4*
     1        (2.214144-K2)+Q5*(3.4596-K2)+Q6*(4.931824-K2)+Q7*(6.780816
     2        -K2)+Q8*(8.856576-K2)+Q9*(11.209104-K2)+Q10*(13.8384-K2)+
     3        Q11*(16.744464-K2)
      I20R3= 2.+K1*I00I+K2*J00R

      GO TO  (420,410,410,390,410,390,380,370,370),ICHUZ

  370 J00I = -K1*(.744*Q1+1.488*Q2+2.232*Q3+2.976*Q4+3.72*Q5+4.464*Q6+
     1             5.208*Q7+5.952*Q8+6.696*Q9+7.44*Q10+8.184*Q11)
      I20I3= -K1*I00R+K2*J00I

      IF  ( ICHUZ .EQ. 8 )                      GO TO 500
  380 I10I = -K1*I00R
  390 I10R = 1.+K1*I00I
```

```
400 GO TO (420,410,410,420,410,410,500,500,500),ICHUZ

410 JOUR = E*(Q1*(.138384-K2+.372*MU*C1)+
   1       E*(Q2*(.553536-K2+.744*MU*C2)+
   2       E*(Q3*(1.245456-K2+1.116*MU*C3)+
   3       E*(Q4*(2.214144-K2+1.488*MU*C4)+
   4       E*(Q5*(3.4596-K2+1.86*MU*C5)+
   5       E*(Q6*(4.981824-K2+2.232*MU*C6)+
   6       E*(Q7*(6.780816-K2+2.604*MU*C7)+
   7       E*(Q8*(8.856576-K2+2.976*MU*C8)+
   8       E*(Q9*(11.209104-K2+3.348*MU*C9)+
   9       E*(Q10*(13.8384-K2+3.72*MU*C10)+
   A       E*(Q11*(16.744464-K2+4.092*MU*C11)))))))))))

     JOUI = -K1*(E*(Q1*(.744+MU*C1) + E*(Q2*(1.488+MU*C2) +
   1             E*(Q3*(2.232+MU*C3)+ E*(Q4*(2.976+MU*C4) +
   2             E*(Q5*(3.72+MU*C5) + E*(Q6*(4.464+MU*C6) +
   3             E*(Q7*(5.208+MU*C7) + E*(Q8*(5.952+MU*C8) +
   4             E*(Q9*(6.696+MU*C9) + E*(Q10*(7.44+MU*C10) +
   5             E*(Q11*(8.184+MU*C11))))))))))))

420 IOUR = .372*E*(R1+E*(2.*R2+E*(3.*R3+E*(4.*R4+E*(5.*R5+E*(6.*R6+
   1           E*(7.*R7+E*(8.*R8+E*(9.*R9+E*(10.*R10+E*11.*R11))))))
   2                 ))))

    IOUI = -K1*(E*(R1+E*(R2+E*(R3+E*(R4+E*(R5+E*(R6+E*(R7+E*(R8+E*(R9
   1           +E*(R10+E*R11)))))))))))
    R1  = R1S
    C6= K1*MU
    C1= SIN(C6)
    C2= COS(C6)
    C3= SQRT(1.+MU*MU)
    C4= MU/C3
    C5= C4/(1.+MU*MU)
    GO TO  (430,440,430,430,440,430,500,500,500),ICHUZ
430 I1UR = C2*(1.-C4+K1*IOUI)-C1*K1*IOUR
    I1UI =-C2*K1*IOUR-C1*(1.-C4+K1*IOUI)
    GO TO  (500,440,440,460,440,440,500,500,500),ICHUZ
440 I2UR3 = C2*(2.*(1.-C4)-C5+K1*IOUI+K2*JOUR)+C1*(C6*(1.-C4)-K1*IOUR
   1          +K2*JOUI)
    I2UI3 = C2*(C6*(1.-C4)-K1*IOUR+K2*JOUI)-C1*(2.*(1.-C4)-C5+K1*IOUI
   1          +K2*JOUR)
    GO TO  (500,500,500,460,450,450,500,500,500),ICHUZ
450 I2UR3 = 2.0 * I2OR3 - I2UR3
    IF  ( ICHUZ-6 )                          500,460,500
460 CAR = 2.*I1OR-I1UR
    I1UR= CAR
500 DK1R=0.
    R1        =   R1S
    DK1I=0.
    DK2R=0.
```

```
      DK2I=0.                                                           TKER 210
      C3=K1*MU1                                                         TKER 211
      C1=COS(C3)                                                        TKER 212
      C2=SIN(C3)                                                        TKER 213
      C3= M*R1/BIGR                                                     TKER 214
      C4=SQRT(1.+MU1*MU1)                                               TKER 215
      C5=KR*X0/BR                                                       TKER 216
      C6=COS(C5)                                                        TKER 217
      C7=SIN(C5)                                                        TKER 218
      GO TO (530,540,530,530,540,530,510,520,510),ICHUZ                 TKER 219
  510 I1UR=I10R                                                         TKER 220
      I1UI=I10I                                                         TKER 221
      IF  ( ICHUZ-7 )                       520,530,520                 TKER 222
  520 I2UR3= I20R3                                                      TKER 223
      I2UI3= I20I3                                                      TKER 224
      IF  ( ICHUZ-8 )                       530,540,530                 TKER 225
  530 CK1R = I1UR + C3*C1/C4                                            TKER 226
      CK1I = I1UI - C3*C2/C4                                            TKER 227
      K10  = 1.0 + X0/BIGR                                              TKER 228
      DK1R = CK1R*C6 + CK1I*C7                                          TKER 229
      DK1I = CK1I*C6 - CK1R*C7                                          TKER 230
      GO TO  (900,540,540,900,540,540,900,540,540),ICHUZ                TKER 231
  540 C8=   (BETA2*(R1/BIGR)**2 + (2.+MU1*C3)/(C4*C4))*(-C3/C4)         TKER 232
      C9= ( K1*C3)*( C3/C4)                                             TKER 233
      CK2R = -I2UR3 + C8*C1 - C9*C2                                     TKER 234
      CK2I = -I2UI3 - C9*C1 - C8*C2                                     TKER 235
      K20  = -2.0 -X0*(2.0+BETA2*(R1/BIGR)**2)/BIGR                     TKER 236
      DK2R = CK2R*C6 + CK2I*C7                                          TKER 237
      DK2I = CK2I*C6 - CK2R*C7                                          TKER 238
  900 KKR = T1*DK1R + T2*DK2R                                           TKER 239
      KKI = T1*DK1I + T2*DK2I                                           TKER 240
      K1RT1  = T1 * DK1R                                                TKER 241
      K1IT1  = T1 * DK1I                                                TKER 242
      K2RT2P = T2P* DK2R                                                TKER 243
      K2IT2P = T2P* DK2I                                                TKER 244
      K10T1  = K10* T1                                                  TKER 245
      K20T2P = K20* T2P                                                 TKER 246
  905 CONTINUE                                                          TKER 247
      KD1R = K1RT1 - K10T1*FLOAT(IND)                                   TKER 248
      KD1I = K1IT1                                                      TKER 249
      KD2R = K2RT2P-K20T2P*FLOAT(IND)                                   TKER 250
      KD2I = K2IT2P                                                     TKER 251
      RETURN                                                            TKER 252
      END                                                               TKER 253
```

```
      SUBROUTINE  TVOR  ( SL1, CL1, TL1, SL2, CL2, TL2, SGS, CGS,        TVOR   2
     *                    SGR, CGR, XO1, XO2, YO , ZO , E  , BETA,       TVOR   3
     *    CBAR,    FMACH,   KR,     BRE,    BIM,                         TVOR   4
     *    IPRNT )                                                        TVOR   5
CTVOR        NORMALWASH AT A POINT (X,Y,Z) - OF A SURFACE DIHEDRAL -     TVOR   6
C            DUE TO A TRAPEZOIDAL UNSTEADY VORTEX RING OF UNIT           TVOR   7
C            STRENGTH.                                                   TVOR   8
C                       SUBROUTINES USED - SNPDF, IDF1, IDF2, FLLD       TVOR   9
C                                                                        TVOR  10
C                                                                        TVOR  11
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *  TVOR  12
C *                                                                   *  TVOR  13
C *       SL1, CL1, TL1  SIN(LAMBDA-1), COS(LAMBDA-1), TAN(LAMBDA-1)  *  TVOR  14
C *       SL2, CL2, TL2  SIN(LAMBDA-2), .....                         *  TVOR  15
C *       SGS, CGS       SIN(GAMMA-S),  ....                          *  TVOR  16
C *       SGR, CGR       SIN(GAMMA-R),  ....                          *  TVOR  17
C *       XO1            X-XI1                                        *  TVOR  18
C *       XO2            X-XI2                                        *  TVOR  19
C *       YO             Y - ETA                                      *  TVOR  20
C *       ZO             Z - ZETA                                     *  TVOR  21
C *       E                                                           *  TVOR  22
C *       BETA           SQRT( 1 -FMACH**2 )                          *  TVOR  23
C *       CV                                                          *  TVOR  24
C *       BR                                                          *  TVOR  25
C *       FMACH          MACH NO.                                     *  TVOR  26
C *       BRE            REAL PART OF B       (RETURNED)              *  TVOR  27
C *       BIM            IMAGINARY PART OF B  (RETURNED)              *  TVOR  28
C *                                                                   *  TVOR  29
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *  TVOR  30
C                                                                        TVOR  31
      REAL     KR, KD1, KD2                                              TVOR  32
C                                                                        TVOR  33
C            VARIABLES DIMENSIONED (2), FIRST WORD IS THE REAL PART OFTVOR  34
C            THE VALUE AND THE SECOND IS THE IMAGINARY PART              TVOR  35
C                                                                        TVOR  36
      DIMENSION      DKI (2),       DKC (2),       DKO (2),              TVOR  37
     *               KD1 (2),       KD2 (2)                              TVOR  38
      DATA  NPOT /6/                                                     TVOR  39
C                                                                        TVOR  40
C                                                                        TVOR  41
      DATA  PI48  / 150.79644720 /                                       TVOR  42
C                                   CALCULATE    BS                      TVOR  43
C                                                                        TVOR  44
      L        =      1                                                  TVOR  45
      CV       =      XO1 - XO2                                          TVOR  46
      SL      =  SL1                                                     TVOR  47
      CL      =  CL1                                                     TVOR  48
      TL      =  TL1                                                     TVOR  49
      XO      =  XO1                                                     TVOR  50
      EE      =  F ** 2                                                  TVOR  51
      TE       =    2.0 * E                                              TVOR  52
      ASSIGN  50   TO  ISNP                                              TVOR  53
```

```
C                                      CALL   SNPDF                          TVOR  54
      GO  TO   1000                                                          TVOR  55
   50 BS          =      DIJ                                                 TVOR  56
      SL      =   SL2                                                        TVOR  57
      CL      =   CL2                                                        TVOR  58
      TL      =   TL2                                                        TVOR  59
      XO      =   XO2                                                        TVOR  60
      ASSIGN   100   TO   ISNP                                               TVOR  61
C                                      CALL SNPDF                            TVOR  62
      GO  TO      1000                                                       TVOR  63
  100 BS          =      BS    -  DIJ                                        TVOR  64
C                                          CALCULATE    DELTA -B             TVOR  65
C                 LIMITS FOR SMALL VALUES OF RADII                           TVOR  66
      EPS         =    0.25 * EE                                             TVOR  67
      IB          =  0                                                       TVOR  68
      IC          =    0                                                     TVOR  69
      FB          =    1.0                                                   TVOR  70
      FC          =    4.0                                                   TVOR  71
C         FIRST CALC.                                                        TVOR  72
C              DELTA-KD- II, IC, AND IO                                      TVOR  73
C                                                                            TVOR  74
      ETL1        =    E * TL1                                               TVOR  75
      ETL2        =    E * TL2                                               TVOR  76
      ESGS        =    E * SGS                                               TVOR  77
      ECGS        =    E * CGS                                               TVOR  78
C                                                                            TVOR  79
      DXO1        =    XO1 + ETL1                                            TVOR  80
      DXO2        =    XO2 + ETL2                                            TVOR  81
      DYO         =    YO  + ECGS                                            TVOR  82
      DZO         =    ZO  + ESGS                                            TVOR  83
C                                                                            TVOR  84
      ASSIGN     200   TO   IFLLD                                            TVOR  85
C                 CALCULATE  R-I  SQUARED AND CALL FLLD IF LARGE ENOUGH      TVOR  86
      R2          =      DYO ** 2 +  DZO ** 2                                TVOR  87
      IF  ( R2   .GE.   EPS )    GO TO 2000                                  TVOR  88
      IB          =      1                                                   TVOR  89
      FC          =    6.0                                                   TVOR  90
      FB          =    0.0                                                   TVOR  91
      GO  TO 230                                                             TVOR  92
  200 DO   220   I = 1, 2                                                    TVOR  93
C                                                                            TVOR  94
  220 DKI(I)     =  KD1(I) / R2  +  KD2(I) / R4                              TVOR  95
C                                   KD1C AND KD2C                            TVOR  96
  230 DXO1        =      XO1                                                 TVOR  97
      DXO2       =   XO2                                                     TVOR  98
      DYO        =   YO                                                      TVOR  99
      DZO        =   ZO                                                      TVOR 100
      ASSIGN     300   TO   IFLLD                                            TVOR 101
C                 CALCULATE  R-C  SQUARED AND CALL FLLD IF LARGE ENOUGH      TVOR 102
      R2          =      DYO ** 2 +  DZO ** 2                                TVOR 103
      IF  ( R2   .GE.   EPS )    GO TO 2000                                  TVOR 104
      IC          =      1                                                   TVOR 105
```

```
      FC         =    0.0
      FB         =    3.0
      GO  TO 330
  300 DO    320    I   = 1, 2
C
  320 DKC(I) =   KD1(I) / R2   +   KD2(I) / R4
C                                KD10 AND KD20
C                     SKIP IF   R-I IS TOO SMALL
  330 IF  ( IB  .NE. 0 ) GO TO 430
      DX01       =   X01 - ETL1
      DX02       =   X02 - ETL2
      DYO        =   YO  - ECGS
      DZO        =   ZO  - ESGS
C
      ASSIGN      400   TO   IFLLD
C                     CALCULATE  R-O  SQUARED AND CALL FLLD IF LARGE ENOUGH
      R2         =       DYO ** 2   +   DZO ** 2
      IF  ( R2   .GE.   EPS )     GO TO 2000
      FB         =       0.0
      FC         =       6.0
      IB         =       1
      GO  TO 430
  400 DO    420   I  = 1, 2
C
  420 DKO(I)     =     KD1(I) / R2    +   KD2(I) / R4
C
  430 COEF       =       1.0 / PI48
      BRE        =       BS/(TE*CV)- COEF*(FB * (DKI(1)+ DKO(1)) + FC*DKC(1))
      BIM        =                  - COFF* ( FB * (DKI(2)+ DKO(2)) + FC*DKC(2))
C
      RETURN
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
 1000 CALL SNPDF ( SL, CL, TL, SGS, CGS, SGR, CGR, XO, YO, ZO, E,
     *                 DIJ, BETA ,   CV  )
C                                  RETURN TO BRANCH
      GO TO    ISNP  ,  ( 50, 100    )
C
C
 2000 CALL   FLLD   ( DX01, DX02, DYC, DZO,     SGR, CGR,   SGS,   CGS,
     *     KR       , CBAR      , FMACH     , E          , L          ,
     *                KD1(1), KD1(2), KD2(1), KD2(2)   )
      IF (   IPRNT   .NE. 0 )   WRITE ( NPOT, 3000)  KD1,   KD2,    R2
 3000 FORMAT( *OKD1=*,2E20.8,* KD2=*, 2E20.8, * R2=*, E20.8)
C
      R4         =       R2 * R2
      GO TO IFLLD , (   200, 30C, 400 )
C
      END
```

```
      SUBROUTINE WANDWT(IPRINT, LINES, NOUT, NTOT, NBOX, NB, NDW, NTAPH,WANDWT 2
     1              NTPDH, NWT, NWTA,                                   WANDWT 3
     2              REFC, KR,  H,  DHDX,   W , DW ,   COL             ) WANDWT 4
C                                                                       WANDWT 5
      REAL          KR                                                  WANDWT 6
      DIMENSION     W(2, NTOT), DW(2, NTOT), COL(2, NTOT),              WANDWT 7
     1              H( 350  ),    DHDX( 350 )                           WANDWT 8
   10 FORMAT   ( 1H1 /// )                                              WANDWT 9
      NWTAP     =   NWT                                                 WANDWT10
      NMODE     =   0                                                   WANDWT11
      IGO       =   1                                                   WANDWT12
      READ    (NTAPH)  NSYM, NASYM                                      WANDWT13
      READ    (NTPDH)                                                   WANDWT14
      DO   15   N = 1, NTOT                                             WANDWT15
      DW(1,N)   =   0.0                                                 WANDWT16
   15 DW(2,N)   =   0.0                                                 WANDWT17
      IF   (IPRINT .NE. 0)  WRITE  (NOUT, 122)                          WANDWT18
      IF   (NB .EQ. 0)   GO TO   70                                     WANDWT19
      REWIND  NDW                                                       WANDWT20
   20 CONTINUE                                                          WANDWT21
      READ     (NDW )   ICOLB                                           WANDWT22
      IF   (ICOLB .EQ. -1)   GO TO   70                                 WANDWT23
      READ     (NDW)   COL                                              WANDWT24
      DO   30   I = 1, NTOT                                             WANDWT25
      DW(1,I)   =   COL(1, I)                                           WANDWT26
      DW(2,I)   =   COL(2, I)                                           WANDWT27
      W (1,I)   =   0.0                                                 WANDWT28
      W (2,I)   =   0.0                                                 WANDWT29
   30 CONTINUE                                                          WANDWT30
   70 CONTINUE                                                          WANDWT31
      READ     (NTAPH)   ICOLP                                          WANDWT32
      READ     (NTPDH)                                                  WANDWT33
      IF   (ICOLP .EQ. -1)   GO TO   150                                WANDWT34
      READ    (NTAPH)   (H(J)       , J = 1, NBOX)                      WANDWT35
      READ    (NTPDH)   (DHDX(J)    , J = 1, NBOX)                      WANDWT36
      DO   90   I = 1, NBOX                                             WANDWT37
      W(1,I)    =   DHDX(I)                                             WANDWT38
      W(2,I)    =   2.0*KR*H(I)  / REFC                                 WANDWT39
   90 CONTINUE                                                          WANDWT40
  100 CONTINUE                                                          WANDWT41
      DO   110   I = 1, NTOT                                            WANDWT42
      W (1,I)   =   -W(1,I) + DW(1,I)                                   WANDWT43
      W (2,I)   =   -W(2,I) + DW(2,I)                                   WANDWT44
  110 CONTINUE                                                          WANDWT45
      WRITE   (NWTAP)    W                                              WANDWT46
C                                                                       WANDWT47
      IF   (IPRINT .EQ. 0)   GO TO   140                                WANDWT48
      IG2       =   0                                                   WANDWT49
      LINE3     =   LINES*3                                             WANDWT50
      IF   (NB .NE. 0)  LINE3 = LINE3 / 2                               WANDWT51
      LPAGE     =   NTOT / LINE3 + 1                                    WANDWT52
      LPAGE     =   1                                                   WANDWT53
```

```
      DO  120  KG = 1, LPAGE                                           WANDWT54
      IG1        =  IG2 + 1                                            WANDWT55
      IG2        =  IG2 + LINE3                                        WANDWT56
      IG2        =  NTOT                                               WANDWT57
      IF  (IG2 .GT. NTOT)  IG2 = NTOT                                  WANDWT58
      IF  (NB .EQ. 0)  GO TO  112                                      WANDWT59
      WRITE  (NOUT,126)  ICOLP, (I, DW(1,I), DW(2,I),  I = IG1, IG2)   WANDWT60
  112 CONTINUE                                                         WANDWT61
      WRITE  (NOUT,130)  ICOLP, (I,  W(1,I),  W(2,I),  I = IG1, IG2)   WANDWT62
      IF  (IG2 .GE. NTOT)  GO TO  120                                  WANDWT63
C                                                                      WANDWT64
      WRITE  (NOUT,10)                                                 WANDWT65
  120 CONTINUE                                                         WANDWT66
C                                                                      WANDWT67
  122 FORMAT  ( 1H1 /// 37H     ***     SYMMETRIC MODES     ***  // )  WANDWT68
  124 FORMAT  ( 1H1 /// 37H     ***   ANTISYMMETRIC MODES   ***  // )  WANDWT69
  126 FORMAT  (     /// 8H  COLUMN, I5, 12H OF  --DW-- //(3(I6,2E13.5)))WANDWT70
  130 FORMAT  (     /// 8H  COLUMN, I5, 12H OF  --WT-- //(3(I6,2E13.5)))WANDWT71
C                                                                      WANDWT72
  140 CONTINUE                                                         WANDWT73
C                                                                      WANDWT74
      IF  (NB .EQ. 0)  GO TO  70                                       WANDWT75
      GO TO  20                                                        WANDWT76
  150 CONTINUE                                                         WANDWT77
      IF  (IGO .EQ. 2 .OR. NASYM .EQ. 0)  GO TO  160                   WANDWT78
      IGO        =  IGO + 1                                            WANDWT79
      NWTAP      =  NWTA                                               WANDWT80
      IF  (IPRINT .NE. 0)  WRITE  (NOUT, 124)                          WANDWT81
      GO TO  140                                                       WANDWT82
  160 CONTINUE                                                         WANDWT83
C                                                                      WANDWT84
      RETURN                                                           WANDWT85
      END                                                              WANDWT86
```

```
      SUBROUTINE ZEROUT( WORK, LENGTH, LOOP, ITAPE )
      COMPLEX    WORK(LENGTH)
C
      DO  10  I = 1, LENGTH
      WORK(I)  =  0.0
   10 CONTINUE
      IF  (ITAPE . EQ . 0)  RETURN
C
      DO  20  L = 1, LOOP
      WRITE  (ITAPE)  (WORK(I),  I = 1, LENGTH)
   20 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE GENMS ( MS, MASS, LIMIT1, LIMIT2,ITM,IMS)
      REAL MS, MASS
      DIMENSION MS (ITM,ITM,1),MASS(IMS,1)
      DIMENSION F(24)
      DO 10 I=1,IMS
   10 F(I)=1.0
      IMSP=IMS+1
      DO 20 I=IMSP,23
   20 F(I)=0.
      DO 60 K = LIMIT1, LIMIT2
      DO 30 I = 1,ITM
      DO 30 J = 1,ITM
   30 MS (I,J,K) = 0.0
      MS (1,1,K) =  MASS (1,K)*F(1)
      MS (2,2,K) =  MASS (1,K)*F(1)
      MS (3,3,K) =  MASS (1,K)*F(1)
      IF( ITM.LT.4)GO TO 40
      MS (2,4,K) = -MASS (4,K)*F(4)
      MS (3,4,K) =  MASS (3,K)*F(3)
      MS (4,4,K) =  MASS (5,K)*F(5)
      IF( ITM.LT.5)GO TO 40
      MS (1,5,K) =  MASS (4,K)*F(4)
      MS (3,5,K) = -MASS (2,K)*F(2)
      MS (4,5,K) = -MASS (8,K)*F(8)
      MS (5,5,K) =  MASS (6,K)*F(6)
      IF( ITM.LT.6)GO TO 40
      MS (1,6,K) = -MASS (3,K)*F(3)
      MS (2,6,K) =  MASS (2,K)*F(2)
      MS (4,6,K) = -MASS (10,K)*F(10)
      MS (5,6,K) = -MASS (9,K)*F(9)
      MS (6,6,K) =  MASS (7,K)*F(7)
      IF( ITM.LT.7)GO TO 40
      MS (1,7,K) =  MASS (15,K)*F(15)
      MS (3,7,K) = -MASS (11,K)*F(11)
      MS (4,7,K) = -MASS (14,K)*F(14)
      MS (5,7,K) =  MASS (12,K)*F(12)
      MS (6,7,K) = -MASS (16,K)*F(16)
      MS (7,7,K) =  MASS (13,K)*F(13)
      IF( ITM.LT.8)GO TO 40
      MS (1,8,K) =  MASS (22,K)*F(22)
      MS (3,8,K) = -MASS (17,K)*F(17)
      MS (4,8,K) = -MASS (21,K)*F(21)
      MS (5,8,K) =  MASS (18,K)*F(18)
      MS (6,8,K) = -MASS (23,K)*F(23)
      MS (7,8,K) =  MASS (19,K)*F(19)
      MS (8,8,K) =  MASS (20,K)*F(20)
   40 CONTINUE
      DO 50 J = 1, ITM
      DO 50 I = J, ITM
   50 MS (I,J,K) = MS (J,I,K)
   60 CONTINUE
      RETURN
```

END GENMS 54

```
      SUBROUTINE INERTM(                                                  INERTM 2
     1 NTI,NMS,EM,ELXIO,ELYIO,ELZIO                                       INERTM 3
     1 , WR, PHIX, PHIY, PHIZ, EMS, AMODNO                                INERTM 4
     2 , CPXDPG,EMPHI,EMBAR,EMWR2,NOMOD,NSYM,NASYM)                       INERTM 5
C                                                                         INERTM 6
      DIMENSION EM(1),ELXIO(1),ELYIO(1),ELZIO(1)                          INERTM 7
      DIMENSION WR(1),PHIX(1),PHIY(1),PHIZ(1),AMODNO(1)                   INERTM 8
      DIMENSION EMS(1)                                                    INERTM 9
      DIMENSION CPXDPG(1),EMPHI(1),EMBAR(1),EMWR2(1)                      INERTM10
C                                                                         INERTM11
      CALL INPUTM(                                                        INERTM12
     1 NTI,NMS,EM,ELXIO,ELYIO,ELZIO                                       INERTM13
     1 , WR,PHIX,PHIY,PHIZ,AMODNO,CPXDPG)                                 INERTM14
C                                                                         INERTM15
      CALL LHS (EM,EMPHI,PHIX,PHIY,PHIZ,NMS,AMODNO,WR                     INERTM16
     1 , EMS, EMBAR, EMWR2, NOMOD, NSYM, NASYM)                           INERTM17
C                                                                         INERTM18
      RETURN                                                              INERTM19
      END                                                                 INERTM20
```

```
      SUBROUTINE INPUTM(NTI,NM
     1 , EM,ELXIO,ELYIO,ELZIO,WR,PHIX,PHIY,PHIZ,AMODNO,CPXDPG)
      DIMENSION EM(1),ELXIO(1),ELYIO(1),ELZIO(1)
      DIMENSION WR(1),PHIX(NM,1),PHIY(NM,1),PHIZ(NM,1)
      DIMENSION AMODNO(20), CPXDPG(1)
      DIMENSION MEMP(20)
C
      COMMON/ZZZ/HEDR(48),NIN,NOUT,KROW,LINES
C
      COMMON /XTF/XF(100)
      EQUIVALENCE (XF(5),NMS)
      EQUIVALENCE (XF(6),NSYM),      (XF(7),NASYM),      (XF(8),NOMOD)
      EQUIVALENCE (XF(80),ISECT),    (XF(81),NDOF),      (XF(82),IMS)
     1 , (XF(96),IPRNTM)
C
      IF  (ISECT.EQ.1) GO TO 200
C
      IF(IPRNTM.EQ.0)GO TO 21
      CALL HEADNG
      WRITE (NOUT,1)
    1 FORMAT(1H0,20X, MASS AND MASS POINT GEOMETRY /1H0)
   21 CONTINUE
C
      DO 90 I=1,NMS
   90 READ  (NTI,310)  EM(I),ELXIO(I),ELYIO(I),ELZIO(I)
      IF(IPRNTM.EQ.0)GO TO 22
      WRITE  (NOUT,480)  (I,EM(I),ELXIO(I),ELYIO(I),ELZIO(I),I=1,NMS)
   22 CONTINUE
C
      DO 100 I=1,NMS
      IF(ELYIO(I).EQ.0)EM(I)=0.50*EM(I)
  100 CONTINUE
C
      IF(IPRNTM.NE.0)CALL HEADNG
C
      DO 190 M=1,NOMOD
C
      IF(IPRNTM.EQ.0)GO TO 23
      WRITE(NOUT,2)M
    2 FORMAT(1H0,20X, FREQUENCY AND INERTIAL MODE SHAPES FOR MODE ,I4)
   23 CONTINUE
C
      READ  (NTI,310)  WR(M)
      IF(IPRNTM.EQ.0)GO TO 24
      LINES=LINES+4
      WRITE  (NOUT,520)  M,WR(M),M,M,M
   24 CONTINUE
      DO 190 I=1,NMS
      READ  (NTI,310)  PHIX(I,M),PHIY(I,M),PHIZ(I,M)
      IF(IPRNTM.EQ.0)GO TO 190
      LINES=LINES+1
      IF(LINES.LT.KROW)GO TO 180
```

```
      CALL HEADNG                                                       INPUTM54
      WRITE(NOUT,2)M                                                    INPUTM55
      WRITE(NOUT,520)M,WR(M),M,M,M                                      INPUTM56
  180 CONTINUE                                                          INPUTM57
C                                                                       INPUTM58
      WRITE(NOUT,530)I,PHIX(I,M),PHIY(I,M),PHIZ(I,M)                    INPUTM59
  190 CONTINUE                                                          INPUTM60
      GO TO 290                                                         INPUTM61
C                                                                       INPUTM62
  200 CALL INPUTS (NTI,NMS,NDOF,IMS,NOMOD,IPRNTM,ELXIO,WR,PHIX,EM)      INPUTM63
C                                                                       INPUTM64
  290 CALL HEADNG                                                       INPUTM65
      READ (NTI,310) (AMODNO(I),I=1,5)                                  INPUTM66
      READ(NTI,310)(AMODNO(I),I=6,9)                                    INPUTM67
      READ (NTI,310) (AMODNO(I),I=11,15)                                INPUTM68
      READ(NTI,310)(AMODNO(I),I=16,20)                                  INPUTM69
      DO 300 I=1,20                                                     INPUTM70
  300 MEMP(I) = AMODNO(I)                                               INPUTM71
      WRITE(NOUT,320)(MEMP(I),I=1,9)                                    INPUTM72
  320 FORMAT(1H0,20X, SYMMETRIC MODE LOCATION DEFINITION /1H0,          INPUTM73
     1 20X, RIGID BODY PLUNGE          MODE ,I3/1H ,                    INPUTM74
     1 20X, RIGID BODY PITCH           MODE ,I3/1H ,                    INPUTM75
     1 20X, RIGID BODY FORE AND AFT    MODE ,I3/1H ,                    INPUTM76
     1 20X, FIRST SYMMETRIC ELASTIC    MODE ,I3/1H ,                    INPUTM77
     1 20X, LAST  SYMMETRIC ELASTIC    MODE ,I3/1H ,                    INPUTM78
     1 20X, SYMMETRIC FLIGHT TRIM      MODE ,I3/1H ,                    INPUTM79
     1 20X, SYMMETRIC AIRCRAFT JIG     MODE ,I3/1H ,                    INPUTM80
     1 20X, FIRST DELETED SYMMETRIC    MODE ,I3/1H ,                    INPUTM81
     1 20X, LAST  DELETED SYMMETRIC    MODE ,I3)                        INPUTM82
      WRITE(NOUT,321)(MEMP(I),I=11,20)                                  INPUTM83
  321 FORMAT(1H0,20X, ANTISYMMETRIC MODE LOCATION DEFINITION /1H0,      INPUTM84
     1 20X, RIGID BODY ROLL            MODE ,I3/1H ,                    INPUTM85
     1 20X, RIGID BODY YAW             MODE ,I3/1H ,                    INPUTM86
     1 20X, RIGID BODY LATERAL         MODE ,I3/1H ,                    INPUTM87
     1 20X, FIRST ANTISYM. ELASTIC     MODE ,I3/1H ,                    INPUTM88
     1 20X, LAST  ANTISYM. ELASTIC     MODE ,I3/1H ,                    INPUTM89
     1 20X, ROLL TRIM                  MODE ,I3/1H ,                    INPUTM90
     1 20X, YAW  TRIM                  MODE ,I3/1H ,                    INPUTM91
     1 20X, ANTISYM AIRCRAFT JIG       MODE ,I3/1H ,                    INPUTM92
     1 20X, FIRST DELETED ANTISYM      MODE ,I3/1H ,                    INPUTM93
     1 20X, LAST  DELETED ANTISYM      MODE ,I3)                        INPUTM94
C                                                                       INPUTM95
   25 CONTINUE                                                          INPUTM96
C                                                                       INPUTM97
      READ (NTI,310) (CPXDPG(I),I=1,NOMOD)                              INPUTM98
      WRITE (NOUT,540) (I,CPXDPG(I),I=1,NOMOD)                          INPUTM99
C                                                                       INPUT100
      RETURN                                                            INPUT101
C                                                                       INPUT102
  310 FORMAT (6F12.0)                                                   INPUT103
430   FORMAT (2X,3H  I,5X,5HEM(I),8X,8HELXIO(I),7X,8HELYIO(I),7X,8HELZIOINPUT104
     1(I)/(I5,1X,E13.6,2X,E13.6,2X,E13.6,2X,E13.6))                     INPUT105
```

```
520    FCRMAT (4H WR(I2,4H) = E11.4/2X,3H  I,3X,8H PHIX(I,I2,1H),5X,7HPHIINPUT106
      1Y(I,I2,1H),5X,7HPHIZ(I,I2,1H))                                     INPUT107
530    FCRMAT (I5,3(2X,E13.6))                                             INPUT108
540    FCRMAT (1H0,3X1HI,4X9HCPXDPG(I)/(1XI4,E13.6))                       INPUT109
       END                                                                INPUT110
```

```
      SUBROUTINE INPUTS (NTI,NMS,NDOF,IMS,NOMOD,IPR,EL,WR,PHI,EM)
C
      DIMENSION EL(NMS,1), WR(1), PHI(NMS,NOMOD,1), EM(IMS,1)
C
      COMMON /ZZZ/HEDR(48),NIN,NOUT,KROW,LINES
      DIMENSION DEFL(8)
      DATADEFL/3H  F,3H  L,3H  H,5HTHETA,5HALPHA,4H PSI,5H BETA,5HDELTA/
C
  310 FORMAT (6F12.0)
  320 FORMAT (1H0,20X,*MODESHAPE FOR MODE*,I4,4X5HFREQ=,E11.4
     1 / 5HOSECT,6X,A5,8(9XA5) )
  321 FORMAT (I5,9E14.6)
C
      DO 110 I=1,NMS
      READ (NTI,310) EL(I,1), EL(I,2), EL(I,3)
      IF (IPR.EQ.0) GO TO 110
      WRITE (NOUT,321) I, (EL(I,J),J=1,3)
  110 CONTINUE
C
      READ (NTI,310) (WR(I),I=1,NOMOD)
C
      DO 130 K=1,NOMOD
      LINES = KROW
      DO 120 J=1,NMS
      READ (NTI,310) (PHI(J,K,I),I=1,NDOF)
      IF (IPR.EQ.0) GO TO 120
      IF (LINES.LT.KROW) GO TO 115
      CALL HEADNG
      WRITE (NOUT,320) K, WR(K), (DEFL(I),I=1,NDOF)
      LINES = LINES+4
  115 WRITE (NOUT,321) J, (PHI(J,K,I),I=1,NDOF)
      LINES = LINES+1
  120 CONTINUE
  130 CONTINUE
C
      DO 140 J=1,NMS
      READ (NTI,310) (EM(I,J),I=1,4)
      READ (NTI,310) (EM(I,J),I=5,10)
      IF (NDOF.LT.7) GO TO 140
      READ (NTI,310) (EM(I,J),I=11,16)
      IF (NDOF.LT.8) GO TO 140
      READ (NTI,310) (EM(I,J),I=17,20)
      READ (NTI,310) (EM(I,J),I=21,23)
  140 CONTINUE
      CALL PRNT (EM,IMS,NMS,IMS,1,16HMASS PROPERTIES ,4)
C
      DO 150 J=1,NMS
      IF (EL(J,2).NE.0) GO TO 150
      DO 145 I=1,IMS
  145 EM(I,J) = 0.5*EM(I,J)
  150 CONTINUE
C
```

```
      RETURN                                                    INPUTS54
      END                                                       INPUTS55
```

```
      SUBROUTINE LHS (FM,EMPHI,PHIX,PHIY,PHIZ,NMS
     1 , AMODNC, FREQ, EMS, CLHS, EMWR2, NEQ, NSYM, NASYM)
C
      COMMON/ZZZ/CASE(48),NIN,NOUT,KROW,LINES,IPRNT,NER
C
      COMMON/XTF/NF(100)
      EQUIVALENCE  (NF(48),IDIMUL)
      EQUIVALENCE (NF(80),ISECT),   (NF(81),NDOF),    (NF(82),IMS)
     1 , (NF(96),IPRNTM)
C
      DIMENSION EM(1), EMPHI(3,NMS,1), EMS(1)
      DIMENSION PHIX(NMS,1),PHIY(NMS,1),PHIZ(NMS,1)
      DIMENSION AMODNC(20)
      DIMENSION CLHS(NEQ,NEQ),FREQ(1),EMWR2(1)
      DO 10 I=1,NEQ
      DO 10 J=1,NEQ
10    CLHS(I,J)=0.0
C
      IF (ISECT.EQ.1) GO TO 50
C
      IF(IDIMUL.EQ.0)GO TO 40
C
      DO 60 IM=1,NMS
      DO 30 J=1,NEQ
      EMPHI(1,IM,J) = EM(IM) * PHIX(IM,J)
      EMPHI(2,IM,J) = EM(IM) * PHIY(IM,J)
      EMPHI(3,IM,J) = EM(IM) * PHIZ(IM,J)
      DO 30 I=1,NEQ
   30 CLHS(I,J) = CLHS(I,J) + PHIX(IM,I) * EMPHI(1,IM,J)
     1        + PHIY(IM,I) * EMPHI(2,IM,J) + PHIZ(IM,I) * EMPHI(3,IM,J)
60    CONTINUE
      GO TO 55
C
 40   CONTINUE
C
      DO 45 I=1,NEQ
      DO 45 J=1,NEQ
      DO 45 K=1,NMS
      CLHS(I,J)=CLHS(I,J)+PHIX(K,I)*PHIX(K,J)*EM(K)
     1                   +PHIY(K,I)*PHIY(K,J)*EM(K)
     1                   +PHIZ(K,I)*PHIZ(K,J)*EM(K)
 45   CONTINUE
      GO TO 55
C
   50 CALL GENMS (EMS,EM,1,NMS,NDOF,IMS)
      IF (IDIMUL.EQ.0) CALL TMST (CLHS,EMS,PHIX,NEQ,NDOF,1,NMS)
      IF (IDIMUL.NE.0) CALL TMSTL (NDOF,1,NMS,NEQ,EMS,CLHS,EMPHI,PHIX)
C
 55   CONTINUE
C
      IF(NSYM.EQ.0.OR.NASYM.EQ.0)GO TO 67
C     NOW ZERO OUT THOSE GM TERMS THAT S/N BE THERE (1/2 A/C ANAL)
```

```
      M2=NSYM+1                                                          LHS
      DO 66 I=1,NSYM                                                     LHS
      DO 66 J=M2,NEQ                                                     LHS
      CLHS(I,J)=0.0                                                      LHS
   66 CLHS(J,I)=0.0                                                      LHS
   67 CONTINUE                                                           LHS
C                                                                        LHS
      DO 65 I=1,NEQ                                                      LHS
65    FMWR2(I) = CLHS(I,I)*(FREQ(I)*6.2831853)**2                        LHS
      NU = NEQ-1                                                         LHS
      DO 90 I=1,NU                                                       LHS
      IF(I.EQ.AMODNO( 6))GO TO 90                                        LHS
      IF(I.EQ.AMODNO( 7))GO TO 90                                        LHS
      IF(I.EQ.AMODNO(16))GO TO 90                                        LHS
      IF(I.EQ.AMODNO(17))GO TO 90                                        LHS
      IF(I.EQ.AMODNO(18))GO TO 90                                        LHS
      KA=I+1                                                             LHS
      DO 80 J=KA,NEQ                                                     LHS
      IF(J.EQ.AMODNO( 6))GO TO 80                                        LHS
      IF(J.EQ.AMODNO( 7))GO TO 80                                        LHS
      IF(J.EQ.AMODNO(16))GO TO 80                                        LHS
      IF(J.EQ.AMODNO(17))GO TO 80                                        LHS
      IF(J.EQ.AMODNO(18))GO TO 80                                        LHS
      IF (ABS(CLHS(I,J))/SQRT(CLHS(I,I)*CLHS(J,J)).LT.1.0E-6) CLHS(I,J)=LHS
     10.0                                                                LHS
80    CONTINUE                                                           LHS
90    CONTINUE                                                           LHS
      DO 100 I=2,NEQ                                                     LHS
      JU=I-1                                                             LHS
      DO 100 J=1,JU                                                      LHS
100   CLHS(I,J)=CLHS(J,I)                                                LHS
C     NOW ZERO OUT THE JIG GEN MASS AND INERTIAL MODE SHAPES IF ANY      LHS
      M1=7                                                               LHS
      M2=7                                                               LHS
  200 DO 230 M=M1,M2                                                     LHS
      IF (AMODNO(M).EQ.0) GO TO 230                                      LHS
      N = AMODNO(M)                                                      LHS
      DO 210 K=1,NEQ                                                     LHS
      CLHS(K,N) = 0.0                                                    LHS
  210 CLHS(N,K) = 0.0                                                    LHS
      DO 220 I=1,3                                                       LHS
      DO 220 J=1,NMS                                                     LHS
  220 FMPHI(I,J,N) = 0.0                                                 LHS
  230 CONTINUE                                                           LHS
      IF (M1.GT.10) GO TO 250                                            LHS
      M1=18                                                              LHS
      M2=18                                                              LHS
      GO TO 200                                                          LHS
  250 CONTINUE                                                           LHS
      CALL HEADNG                                                        LHS
      CALL PRNT(CLHS,NEQ,NEQ,NEQ,1,                                      LHS
     1 24HGENERALIZED MASS MATRIX  ,6)                                   LHS
```

```
      CALL  PRNT(EMWR2,NEQ,1,NEQ, 1,                                    LHS  106
     1 28HGENERALIZED STIFFNESS MATRIX        ,7)                       LHS  107
C                                                                       LHS  108
      RETURN                                                            LHS  109
      END                                                               LHS  110
```

```
      SUBROUTINF TMST ( EMBAR, MS, TM, NMODES, ITM, NFTMB, NLTMB)
      REAL MS
      DIMENSION MS (ITM, ITM, 1 ), TM(NLTMB,NMODES,1)
      DIMENSION FMBAR(NMODES,NMODES), TEMP(8)
      DO 137 K = NFTMB, NLTMB
      DO 133 J = 1, NMODES
      DO 131 I = 1, ITM
      TEMP (I) = 0.0
      DO 131 L = 1, ITM
  131 TEMP (I) = MS (I,L,K) * TM (K,J,L) + TEMP (I)
      DO 133 I = J, NMODES
      DO 133 L = 1, ITM
  133 EMBAR(I,J) = TM(K,I,L) * TEMP (L) + EMBAR (I,J)
      DO 137 J = 1, NMODES
      DO 137 I = J, NMODES
  137 EMBAR (J,I) = EMBAR (I,J)
      RETURN
      FND
```

```
      SUBROUTINE TMSTL(ITM,NFTMB,NLTMB,NM,MS,EMBAR,EMPHI,TM)
      REAL MS
      DIMENSION MS(ITM,ITM,NLTMB)
      DIMENSION EMBAR(NM,NM)
      DIMENSION EMPHI(ITM,NLTMB,NM)
      DIMENSION TM(NLTMB,NM,ITM)
      DO 10 J=1,ITM
      DO 10 K=1,NM
      DO 10 I=NFTMB,NLTMB
      EMPHI(J,I,K)=0.
      DO 10 L=1,ITM
   10 EMPHI(J,I,K)=EMPHI(J,I,K)+MS(J,L,I)*TM(I,K,L)
      DO 30 I=1,NM
      DO 30 J=I,NM
         DO 20 L=1,ITM
      DO 20 K=NFTMB,NLTMB
   20 EMBAR(I,J)=EMBAR(I,J)+TM(K,I,L)*EMPHI(L,K,J)
   30 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE LOAD(
     1   NTI, ELXIO, FLYIO, ELZIO, FMPHI, NDOF, NMS, NSYM, NASYM,
     1 PHIX,PHIY,PHIZ,NENGS,NENGM)
C
      DIMENSION FLXIO(1), ELYIO(1), FLZIO(1)
      DIMENSION FMPHI(NDOF,NMS,1)
      DIMENSION PHIX(NMS,1),PHIY(NMS,1),PHIZ(NMS,1),NENGM(1)
      DIMENSION IHD(50)
C
      COMMON NAA,A(1)
      COMMON /ZZZ/HEDR(48),NIN,NOUT,KROW,LINES,IPRNT,NER
      COMMON /XTF/XF(100)
      EQUIVALENCE (XF(11),IDENT)
      EQUIVALENCE (XF(31),NBEAMS),   (XF(32),NINTLD), (XF(33),NSTRSS)
     1     ,         (XF(34),NMGRP),    (XF(35),NABGRP), (XF(36),NSBGRP)
     2     ,         (XF(37),NBOXES),   (XF(38),NAERSB)
     3     ,         (XF(41),NK),       (XF(42),NG)
     4     ,         (XF(43),NBOX),     (XF(44),NSBETO)
     5 , (XF(24),KPRLDS),(XF(8),NOMOD)
     6     ,         (XF(80),ISECT)
      COMMON /DDTBLS/DDTBL(20,10)
      INTEGER DDTBL
      DATA CORERQ/8HUNITLOAD/
C
  999 FORMAT(1H0,20X,I10,  WORDS OF CORE RQD FOR STEP +++ ,A10, +++ )
C
      NM = NSYM + NASYM
C
      L1  = NAA
      L2   = L1   + NBEAMS*7
      L3   = L2   + NINTLD*8
      L4   = L3   + 3*NMGPP
      L5   = L4   + 3*NABGRP
      L6   = L5   + 3*NSBGRP
      L7   = L6   + NSTRSS*NINTLD
C
      CALL INPUTL (NTI,A(L1),NBEAMS,A(L2),NINTLD,A(L6),NSTRSS
     1 , A(L3),NMGPP,A(L4),NABGRP,A(L5),NSBGRP,NMASS,NBOXES,NAERS3
     2 , A(L7),NENGS,NENGM)
C
      IHD(2) = IDENT
      IHD(3) = NINTLD
      IHD(4) = NSTRSS
      IHD(5) = NENGS
      IHD(6) = NSYM
      IHD(7) = NASYM
      IHD(8) = NK
      IHD(9) = NABGRP
      IHD(10) = NSBGRP
      IHD(11) = NBOXES
      IHD(12) = NAERSB
      DO 100 I=13,50
```

```
  100 IHD(I) = 0
      CALL WRUNIT (1,0,IHD,NOUT,NER)
C     WRITE STALDS AND STRESS ON NTU
      CALL WRUNIT (2,0,A(L2),NOUT,NER)
      CALL WRUNIT (3,0,A(L6),NOUT,NER)
C
      CALL RTOI (A(L3),A(L3),3*(NMGRP+NABGRP+NSBGRP))
C
      L8  = L7  + 3*3*NMGRP
      L9  = L8  + NBEAMS*NBEAMS
      L91 = L9 + NBEAMS
      L92 = L91 + NINTLD
      L10 = L92 + NINTLD
      L11 = L10 + 3*3*NBEAMS
C
      CALL BEAM (NBEAMS,A(L1),A(L8),A(L9),A(L10),A(L11) )
C
      CALL SYMSOL(NINTLD,A(L91),A(L92),A(L2),A(L10))
C
C
      L12 = L11 + 4*NBEAMS
      LENGTH  =   NINTLD*NMGRP*NMASS
      L13 = L12 + LENGTH
      L14 = L13 + LENGTH
      L15 = L14 + LENGTH
      IF (ISECT.EQ.0) GO TO 110
      L141  = L15
      L142  = L141 + LENGTH
      L143  = L142 + LENGTH
      L15   = L143 + LENGTH
  110 CONTINUE
      L16 = L15 + NINTLD*NENGS
      L17 = L16 + NSYM*NENGS
C
      LMAX = MAX0(L17,L15+NINTLD*NM)
C
      WRITE(NOUT,999)LMAX,CORERQ
C
      CALL GEOMMB (NMGRP,A(L3),NBEAMS,A(L9),A(L8),NINTLD,A(L2)
     1 , A(L12), A(L13), A(L14), A(L141), A(L142), A(L143)
     2 , FLXIO, FLYIO, FLZIO, A(L11), A(L10), A(L7) )
C
      IF(NENGS.NE.0)CALL THRUST(
     1 NENGS,NMS,NMGRP,NINTLD,NSYM,NENGM,ELXIO,ELYIO,ELZIO,
     1 PHIX,PHIY,PHIZ,A(L12),A(L13),A(L14),A(L3),A(L7),A(L15),A(L16) )
C
C     WRITE THRLOD AND THRGNF ON UNIT LOAD TAPE HERE
C
      CALL WRUNIT (4,0,A(L15),NOUT,NER)
      CALL WRUNIT (5,0,A(L16),NOUT,NER)
C
      IF(KPRLDS.EQ.0)GO TO 210
```

```
C
      CALL HEADNG
      DO 200 L=1,NOMOD
      WRITE(NOUT,190)L
  190 FORMAT(1H ,20X, MASS TIMES MODES, MODE ,I4)
      CALL PRNT (FMPHI(1,1,L), NDOF, NMS, NDOF,1,16HMPHI(NDOF,NMS)    ,3)
  200 CONTINUE
C
  210 CALL MSSPHI (NMGRP, A(L3), NINTLD
     1 , A(L12), A(L13), A(L14), A(L141), A(L142), A(L143)
     2 , NDOF,NMS,FMPHI,NM,A(L15), A(L91), A(L92), NSYM,NASYM )
C
C     WRITE PIQ ON NTU
C
      CALL WRUNIT (6,0,A(L15),NOUT,NER)
C
      LENGTH  =   NINTLD*NABGRP*NBOXES
      L13 = L12 + LENGTH
      LENGTH = NINTLD*NSBGRP*NAERSB
      L14 = L13 + LENGTH
      L15 = L14 + LENGTH
      L16 = L15 + NBOXES
      L17 = L16 + NBOXES
      L18 = L17 + NBOXES
      CALL RDAERO (4HGEO1,0,A(L18),NOUT,NER)
      IF (NER.NE.0) GO TO 1000
      L19 = L18 + NBOX
      L20 = L19 + NBOX
      L21 = L20 + NBOX
      L22 = L21 + NBOX
      L23 = L22 + NBOX
      L24 = L23 + NBOX
C
      WRITE(NOUT,999)L24,CORERQ
C
      CALL GEOMAB (NABGRP,A(L4),NBEAMS,A(L9),A(L8),NINTLD,A(L2)
     1 , A(L12),A(L15),A(L16),A(L17),A(L11),A(L10)
     2 , A(L18),A(L19),A(L20),A(L21),A(L22),A(L23) )
C
      L16 = L15 + NAERSB
      L17 = L16 + NAERSB
      L18 = L17 + NAERSB
      CALL RDAERO (4HGEOS,0,A(L18),NOUT,NER)
      IF (NER.NE.0) GO TO 1000
      L19 = L18 + NSBETO
      L20 = L19 + NSBETO
      L21 = L20 + NSBETO
      L22 = L21 + NSBETO
      L23 = L22 + NSBETO
      L24 = L23 + NSBETO
C
      WRITE(NOUT,999)L24,CORERQ
```

```
C                                                                       LOAD 158
      CALL GEOMSB (NSBGRP,A(L5),NBEAMS,A(L9),A(L8),NINTLD,A(L2)         LOAD 159
     1 , A(L13),A(L14),A(L15),A(L16),A(L17),A(L11),A(L10)               LOAD 160
     2 , A(L18),A(L19),A(L20),A(L21),A(L22),A(L23) )                    LOAD 161
C                                                                       LOAD 162
      MXROW = MAXO(NBOX,NSBETC)                                         LOAD 163
      MXCOL=MAXO(NSYM,NASYM)                                            LOAD 164
      L16 = L15 + 2*MXROW*MXCOL                                         LOAD 165
      L17 = L16 + 2*NINTLD*MXCOL                                        LOAD 166
C                                                                       LOAD 167
      WRITE(NOUT,999)L17,COREFRQ                                        LOAD 168
      CALL AROLOD (NINTLD,A(L16),NSYM,NASYM,NK,NABGRP                   LOAD 169
     1 , A(L4),NSBGRP,A(L5),A(L12),A(L13),A(L14),A(L15),                LOAD 170
     2  A(L91),A(L92))                                                  LOAD 171
C                                                                       LOAD 172
 1000 RETURN                                                            LOAD 173
      END                                                               LOAD 174
```

```
      SUBROUTINE ABAMLT(LOADCD,PINTP,TLAMM,DX,DY,DZ,GAMZ,GAMY)          ABAMLT 2
C                                                                       ABAMLT 3
      DIMENSION TLAMM(3,3)                                              ABAMLT 4
C                                                                       ABAMLT 5
      GO TO (501,502,503,504,505,506),LOADCD                            ABAMLT 6
C                                                                       ABAMLT 7
  501 AL=-TLAMM(1,1)*DZ+TLAMM(1,3)*DX                                   ABAMLT 8
      AH=TLAMM(1,1)*DY-TLAMM(1,2)*DX                                    ABAMLT 9
      GO TO 510                                                         ABAMLT10
C                                                                       ABAMLT11
  502 AL=-TLAMM(2,1)*DZ+TLAMM(2,3)*DX                                   ABAMLT12
      AH=TLAMM(2,1)*DY-TLAMM(2,2)*DX                                    ABAMLT13
      GO TO 510                                                         ABAMLT14
C                                                                       ABAMLT15
  503 AL=-TLAMM(3,1)*DZ+TLAMM(3,3)*DX                                   ABAMLT16
      AH=TLAMM(3,1)*DY-TLAMM(3,2)*DX                                    ABAMLT17
      GO TO 510                                                         ABAMLT18
C                                                                       ABAMLT19
  504 AL=TLAMM(3,2)                                                     ABAMLT20
      AH=TLAMM(3,3)                                                     ABAMLT21
      GO TO 510                                                         ABAMLT22
C                                                                       ABAMLT23
  505 AL=TLAMM(2,2)                                                     ABAMLT24
      AH=TLAMM(2,3)                                                     ABAMLT25
      GO TO 510                                                         ABAMLT26
C                                                                       ABAMLT27
  506 AL=TLAMM(1,2)                                                     ABAMLT28
      AH=TLAMM(1,3)                                                     ABAMLT29
C                                                                       ABAMLT30
C                                                                       ABAMLT31
  510 CONTINUE                                                          ABAMLT32
C                                                                       ABAMLT33
C     NOW ROTATE TO AERO LOAD SYSTEM                                    ABAMLT34
C                                                                       ABAMLT35
      PINTP=-AL*GAMY+AH*GAMZ                                            ABAMLT36
C                                                                       ABAMLT37
C                                                                       ABAMLT38
      RETURN                                                            ABAMLT39
      END                                                               ABAMLT40
```

```
      SUBROUTINE AROLOD (NINTLD,PAQ,NSYM,NASYM,NK,NABGRP
     1 ,NFNLAB,NSBGRP,NFNLSB,PINTP,PINTZ,PINTY,FPH,SYMCOD,ASMCOD)
C
      COMMON/ZZZ/HEDR(48),NIN,NOUT,KROW,LINES
C
      COMMON/XTF/NF(100)
      EQUIVALENCE (NF(24),KPRCHK),   (NF(42),NGUST)
     1       ,      (NF(43),NBOX),     (NF(44),NSBETO)
C
      COMPLEX FPH(1)
      DIMENSION PAQ(2,NINTLD,1)
      DIMENSION NFNLAB(3,1)
      DIMENSION NFNLSB(3,1)
      DIMENSION PINTP(NINTLD,NABGRP,1)
      DIMENSION PINTZ(NINTLD,NSBGRP,1)
      DIMENSION PINTY(NINTLD,NSBGRP,1)
      DIMENSION SYMCOD(1),ASMCOD(1)
      DIMENSION AERO(6)
      DATA AERO/4HDPSP,4HDZSP,4HDYSP,4HDPAP,4HDZAP,4HDYAP/
C
C
    1 FORMAT(1H ,20X, SYMMETRIC AERO FOR RED FREQ NO ,I4)
    2 FORMAT(1H ,20X, ANTISYMMETRIC AERO FOR RED FREQ NO ,I4)
C
C     CALC INTGD AERO AND GUST LOADS DUE TO UNIT MODAL AMPLITUDE,GUST VE
C
      DO 1000 K=1,NK
      IA = 1
      NLOOP=1
      NMT=NSYM
C
  100 CONTINUE
C
      NN1=2*NINTLD*NMT
      DO 110 I=1,NN1
  110 PAQ(I,1,1)=0.
C
C
C     READ DPOS OR DPOA INTO FPH
      CALL RDAERO (AERO(IA),K,FPH,NOUT,NER)
C
      CALL AROPHA (NABGRP,NFNLAB,NINTLD,PINTP,NBOX,FPH,NMT,PAQ)
C
      IF(KPRCHK.EQ.0)GO TO 400
      CALL PRNT(FPH,NBOX,NMT,NBOX,2,12HDPOS OR DPOA  ,3)
C
      IF(NLOOP.EQ.1)WRITE(NOUT,1)K
      IF(NLOOP.GT.1)WRITE(NOUT,2)K
      CALL PRNT( PAQ,NINTLD,NMT,NINTLD,2,12HPAQS OR PAQA ,3)
C
  400 CONTINUE
C
```

```
C     READ DZOS OR DZCA INTO FPH
      IA = IA+1
      CALL RDAERO (AERO(IA),K,FPH,NOUT,NER)
C
      CALL AROPHA(NSBGRP,NFNLSB,NINTLD,PINTZ,NSBETO,FPH,NMT,PAQ)
C
      IF(KPRCHK.EQ.0)GO TO 410
      CALL PRNT(FPH,NSBETO,NMT,NSBETO,2,12HDZOS OR DZOA ,3)
C
      IF(NLOOP.EQ.1)WRITE(NOUT,1)K
      IF(NLOOP.GT.1)WRITE(NOUT,2)K
      CALL PRNT( PAQ,NINTLD,NMT,NINTLD,2,12HPAQS OR PAQA ,3)
C
  410 CONTINUE
C
C     READ DYOS OR DYCA INTO FPH
      IA = IA+1
      CALL RDAERO (AERO(IA),K,FPH,NOUT,NER)
C
      CALL AROPHA(NSBGRP,NFNLSB,NINTLD,PINTY,NSBETO,FPH,NMT,PAQ)
C
      IF(KPRCHK.EQ.0)GO TO 420
      CALL PRNT(FPH,NSBETO,NMT,NSBETO,2,12HDYOS OR DYOA ,3)
C
      IF(NLOOP.EQ.1)WRITE(NOUT,1)K
      IF(NLOOP.GT.1)WRITE(NOUT,2)K
      CALL PRNT( PAQ,NINTLD,NMT,NINTLD,2,12HPAQS OR PAQA ,3)
C
  420 CONTINUE
C
C
C     NOW CLEAN UP FOR C/L INTGD LOADS
C
      DO 325 L=1,NINTLD
      IF(NLOOP.EQ.1)SCODE=SYMCOD(L)
      IF(NLOOP.EQ.2)SCODE=ASMCOD(L)
      DO 325 M=1,NMT
      PAQ(1,L,M)=PAQ(1,L,M)*SCODE
  325 PAQ(2,L,M)=PAQ(2,L,M)*SCODE
C
      IF(KPRCHK.NE.0)CALL PRNT(PAQ,NINTLD,NMT,NINTLD,2,
     1 32HPAQS OR PAQA MODED FOR C/L LOADS   ,8)
C
C     NOW SAVE PAQS OR PAQA ON NTU
C
      IR = NLOOP+6
      CALL WRUNIT (IR,K,PAQ,NOUT,NER)
C
C     LOOP BACK FOR ANTISYM AERO IF NOT COMPLETE
C
      NLOOP=NLOOP+1
      IF(NLOOP.GT.2)GO TO 1000
```

```
      IF (NASYM.EQ.0) GO TO 1000                                        AROL0106
      IA = 4                                                            AROL0107
      NMT=NASYM                                                         AROL0108
      GO TO 100                                                         AROL0109
C                                                                       AROL0110
 1000 CONTINUE                                                          AROL0111
C                                                                       AROL0112
C     WRITE SYMCOD AND ASMCOD ON UNIT LOAD TAPE                         AROL0113
C                                                                       AROL0114
      CALL WRUNIT(9 ,0,SYMCOD,NOUT,NER)                                 AROL0115
      CALL WRUNIT(10,0,ASMCOD,NOUT,NER)                                 AROL0116
C                                                                       AROL0117
C     NOW SAVE NFNLAB AND PINTP ON NTU                                  AROL0118
C                                                                       AROL0119
      CALL WRUNIT (11,0,NFNLAB,NOUT,NER)                                AROL0120
      CALL WRUNIT (12,0,PINTP,NOUT,NER)                                 AROL0121
C                                                                       AROL0122
C     NOW SAVE NFNLSB AND PINTY AND PINTZ ON NTU                        AROL0123
C                                                                       AROL0124
      CALL WRUNIT (13,0,NFNLSB,NOUT,NER)                                AROL0125
      CALL WRUNIT (14,0,PINTZ,NOUT,NER)                                 AROL0126
      CALL WRUNIT (15,0,PINTY,NOUT,NER)                                 AROL0127
C                                                                       AROL0128
      RETURN                                                            AROL0129
      END                                                               AROL0130
```

```
      SUBROUTINE AROPHA(NGRP,NFNL,NINTLD,PINT,NROW,FPH,NCOL,P)
C
      DIMENSION P(2,NINTLD,1)
      DIMENSION PINT(NINTLD,NGRP,1)
      DIMENSION FPH(2,NROW,1)
      DIMENSION NFNL(3,1)
C
C
      DO 100 M=1,NINTLD
      DO 100 N=1,NCOL
      DO 100 L=1,NGRP
      NAFRO=NFNL(2,L)-NFNL(1,L)+1
      DO 100 J=1,NAFRO
      LOC=NFNL(1,L)+J-1
      P(1,M,N)=P(1,M,N)+PINT(M,L,J)*FPH(1,LOC,N)
      P(2,M,N)=P(2,M,N)+PINT(M,L,J)*FPH(2,LOC,N)
  100 CONTINUE
C
C
      RETURN
      END
```

```
      SUBROUTINE BEAM (NBEAMS,BEAMGM,NBSEQ,NBCON,TLAMM,TLAMY)
C
C
      DIMENSION BEAMGM(NBEAMS,7)
      DIMENSION TLAMM(3,3,1), TLAMY(4,1)
      DIMENSION NBSEQ(NBEAMS,1),NBCON(NBEAMS)
C
      COMMON /ZZZ/HEDR(49),NOUT,KROW,LINES,IPRNT,NER
C
      DATA EPS/.001/
C
      WRITE(NOUT,1)
  1   FORMAT(1H1,20X,*BEAM LOAD DIST DEFINITION*/1H0,
     1 *BEAM  COSDIH  SINDIH  COSSWP  SINSWP    BEAMS LOADED BY NO.*)
C
C
C     THIS LOOP CALCULATES THE LOAD SEQUENCE THRU THE BEAM NETWORK
C     FIND ALL BEAMS CONNECTED TO IB END OF ITH BEAM
C
C
      DO 200 I=1,NBEAMS
      NCB=1
      M=I
      NBSEQ(I,NCB)=I
C
 102  DO 101 J=1,NBEAMS
      N=J
      IF(J.EQ.I)GO TO 101
      DX=BEAMGM(J,4)-BEAMGM(M,1)
      IF(ABS(DX).GT.EPS)GO TO 101
      DY=BEAMGM(J,5)-BEAMGM(M,2)
      IF(ABS(DY).GT.EPS)GO TO 101
      DZ=BEAMGM(J,6)-BEAMGM(M,3)
      IF(ABS(DZ).GT.EPS)GO TO 101
      NCB=NCB+1
      NBSEQ(I,NCB)=N
      GO TO 103
C
 101  CONTINUE
      GO TO 100
C
 103  M=N
      GO TO 102
C
 100  NBCON(I)=NCB
C
C
C     THIS PART CALCULATES DIRECTION COSINE MATRICES FOR INTEGRATED LOAD
C     ROTATION FROM AAS SYSTEM TO LOCAL BEAM SYSTEMS
C
      DX=BEAMGM(I,4)-BEAMGM(I,1)
      DY=BEAMGM(I,5)-BEAMGM(I,2)
```

```
      DZ=BEAMGM(I,6)-BEAMGM(I,3)
      IF(ABS(DX).LT.EPS)DX=0.
      IF(ABS(DY).LT.EPS)DY=0.
      IF(ABS(DZ).LT.EPS)DZ=0.
      AL=SQRT(DX**2+DY**2+DZ**2)
      DDL=SQRT(DY**2+DZ**2)
C
      COSSWP=DDL/AL
      SINSWP=DX/AL
      IF(DDL.LT.EPS)GO TO 105
      COSDIH=DY/DDL
      SINDIH=DZ/DDL
      GO TO 106
C
  105 COSDIH=0.
      SINDIH=1.
C
  106 CONTINUE
C
C     FORM DIRECTION COSINE MATRIX, XBEAM=R*XAAS
C
      R11=COSSWP
      R12=-SINSWP*COSDIH
      R13=-SINSWP*SINDIH
      R21=SINSWP
      R22=COSSWP*COSDIH
      R23=COSSWP*SINDIH
      R31=0.
      R32=-SINDIH
      R33=COSDIH
C
C     FORM DIR.COS. FOR YBEAM IN TERMS OF X,Y,Z AAS
C
      TLAMY(1,I)=R21
      TLAMY(2,I)=R22
      TLAMY(3,I)=R23
C
C     FIND ORIGIN OF BEAM IN AAS SYSTEM
C
      TLAMY(4,I)=-(R21*BEAMGM(I,1)+R22*BEAMGM(I,2)+R23*BEAMGM(I,3))
C
      L=BEAMGM(I,7)
C
C     FORM INTG D LOAD DIR.COS IN TERMS OF AAS LOADS,BY BEAM COMP. NO.
C     COMP.SEQ. IS        1=WING OR HORIZONTAL TAIL
C                          2=FUSELAGE
C                         3=VERTICAL TAIL
C                          4=WING PODS
C                         5=FUSELAGE PODS
C                         6=VERTICAL STAB. PODS
C
      TLAMM(1,1,I) = R11
```

```
      TLAMM(1,2,I) = R12                                                BEAM 106
      TLAMM(1,3,I) = R13                                                BEAM 107
      TLAMM(2,1,I) = R21                                                BEAM 108
      TLAMM(2,2,I) = R22                                                BEAM 109
      TLAMM(2,3,I) = R23                                                BEAM 110
      TLAMM(3,1,I) = R31                                                BEAM 111
      TLAMM(3,2,I) = R32                                                BEAM 112
      TLAMM(3,3,I) = R33                                                BEAM 113
C                                                                       BEAM 114
      K=NBCON(I)                                                        BEAM 115
      WRITE(NOUT,2)I,COSDIH,SINDIH,COSSWP,SINSWP,                       BEAM 116
     1 (NBSEQ(I,J),J=1,K)                                               BEAM 117
    2 FORMAT (1H0,I4,4F10.6,4X,20I3/10(1H ,48X,20I3/))                  BEAM 118
C                                                                       BEAM 119
  200 CONTINUE                                                          BEAM 120
C                                                                       BEAM 121
C                                                                       BEAM 122
      CALL PRNT (TLAMY,4,NBEAMS,4,1,5HTLAMY,2)                          BEAM 123
      DO 250 K=1,NBEAMS                                                 BEAM 124
      WRITE (6,240) K                                                   BEAM 125
  240 FORMAT (1H0,6HNBEAM=,I3)                                          BEAM 126
      LINES = LINES+2                                                   BEAM 127
  250 CALL PRNT (TLAMM(1,1,K),3,3,3,1,5HTLAMM,2)                        BEAM 128
      RETURN                                                            BEAM 129
      END                                                               BEAM 130
```

```
      SUBROUTINE GEOMAB(NABGRP,NFNLBB,NBEAMS,NBCON,NBSEQ,NINTLD,STALDS,
     1 FINTP,BBXIO,BBYIO,BBZIO,TLAMY,TLAMM,
     1 XI1,ETA1,ZETA1,XI2,ETA2,ZETA2)
C
C     THIS SUBROUTINE CALCS THE INTEGRATED LOADS IN TERMS OF ALL
C     UNIT  AERO LOADS AND THEN IN TERMS OF GEN. RESPONSE
C
C
      DIMENSION NFNLBB(3,1)
      DIMENSION NBCON(1)
      DIMENSION NBSEQ(NBEAMS,1)
      DIMENSION STALDS(NINTLD,8)
      DIMENSION PINTP(NINTLD,NABGRP,1)
      DIMENSION BBXIO(1),BBYIO(1),BBZIO(1)
      DIMENSION TLAMY(4,1)
      DIMENSION TLAMM(3,3,1)
      DIMENSION XI1(1),XI2(1)
      DIMENSION ETA1(1),ETA2(1)
      DIMENSION ZETA1(1),ZETA2(1)
C
      COMMON /ZZZ/HFDR(49),NOUT,KROW,LINES,IPRNT,NER
      COMMON/XTF/NF(100)
      EQUIVALENCE (NF(24),KPRLDS)
C
C     LOOP THRU ALL AERO BOX GROUPS
C
      DO 500 L=1,NABGRP
      NBOXES=NFNLBB(2,L)-NFNLBB(1,L)+1
      NFBEAM=NFNLBB(3,L)
      KK=NBCON(NFBEAM)
      DO 40 M=1,NINTLD
      DO 40 J=1,NBOXES
   40 PINTP(M,L,J) = 0.0
C
       WRITE(NOUT,8)L
  8   FORMAT(1H1,20X,*BOX LOAD DATA*/1HO,20X,
     1  *GEOMETRY FOR PANEL*,I4/1HO,* BOX*,
     1  6X,*X*,11X,*Y*,11X,*Z*)
C
 50   CONTINUE
C
C
C     FIND DIR COS AND BOX FORCE COORDS
      LOC=NFNLBB(1,L)
      DX=XI2(LOC)-XI1(LOC)
      DY=ETA2(LOC)-ETA1(LOC)
      DZ=ZETA2(LOC)-ZETA1(LOC)
      DDL=SQRT(DY**2+DZ**2)
      SINDIH=DZ/DDL
      COSDIH=DY/DDL
      GAMY=SINDIH
      GAMZ=COSDIH
```

```
      DO 100 J=1,NBOXES                                                GEOMAB54
      LOC=NFNLBB(1,L)+J-1                                              GEOMAB55
      BBXIO(J)=(XI2(LOC)+XI1(LOC))/2.                                  GEOMAB56
      BBYIO(J)=(ETA2(LOC)+ETA1(LOC))/2.                                GEOMAB57
      BBZIO(J)=(ZETA2(LOC)+ZETA1(LOC))/2.                              GEOMAB58
C                                                                      GEOMAB59
      WRITE(NOUT,7)LOC,BBXIO(J),BBYIO(J),BBZIO(J)                      GEOMAB60
    7 FORMAT (1H ,I4,3E14.6)                                           GEOMAB61
C                                                                      GEOMAB62
C                                                                      GEOMAB63
 100  CONTINUE                                                         GEOMAB64
C                                                                      GEOMAB65
      IF(KPRLDS.EQ.0)GO TO 105                                         GEOMAB66
      WRITE(NOUT,6)                                                    GEOMAB67
 6    FORMAT(1H0,20X,*INTEGRATED LOADS PER BOX LOAD*/                  GEOMAB68
     1  1H0,*LOAD*,20X,*BOX NO.*)                                      GEOMAB69
C                                                                      GEOMAB70
 105  CONTINUE                                                         GEOMAB71
C      FIND ALL BEAMS THIS AERO BOX GROUP LOADS UP                     GEOMAB72
C                                                                      GEOMAB73
      DO 400 K=1,KK                                                    GEOMAB74
      NBEAM=NBSEQ(NFBEAM,K)                                            GEOMAB75
C                                                                      GEOMAB76
C     FIND ALL INTEGRATED LOAD STA. THIS AERO BOX GROUP LOADS UP       GEOMAB77
C                                                                      GEOMAB78
      DO 300 M=1,NINTLD                                                GEOMAB79
      NB=STALDS(M,1)                                                   GEOMAB80
      IF (NB.NE.NBEAM) GO TO 300                                       GEOMAB81
C                                                                      GEOMAB82
      XB=STALDS(M,4)                                                   GEOMAB83
      YB=STALDS(M,5)                                                   GEOMAB84
      ZB=STALDS(M,6)                                                   GEOMAB85
      LOADCD=STALDS(M,2)                                               GEOMAB86
C                                                                      GEOMAB87
C     NOW CALC INT.LOADS FOR ALL UNIT  AERO LOADS IN THIS AERO BOX GRO GEOMAB88
C                                                                      GEOMAB89
      DO 200 J=1,NBOXES                                                GEOMAB90
      XM=BBXIO(J)                                                      GEOMAB91
      YM=BBYIO(J)                                                      GEOMAB92
      ZM=BBZIO(J)                                                      GEOMAB93
      DX=XM-XB                                                         GEOMAB94
       DY=YM-YB                                                        GEOMAB95
      DZ=ZM-ZB                                                         GEOMAB96
C                                                                      GEOMAB97
C     TEST FOR A AERO BOX IB OF THIS BEAM LOAD STA. AND DON T USE      GEOMAB98
C                                                                      GEOMAB99
      IF(NB.NE.NFBEAM)GO TO 150                                        GEOMA100
      YLOAD=TLAMY(1,NFBEAM)*XM                                         GEOMA101
     1 +TLAMY(2,NFBEAM)*YM                                             GEOMA102
     1 +TLAMY(3,NFBEAM)*ZM                                             GEOMA103
     1 +TLAMY(4,NFBEAM)                                                GEOMA104
      YBEAM=TLAMY(1,NFBEAM)*XB                                         GEOMA105
```

```
     1 +TLAMY(2,NFBEAM)*YB                                                    GEOMA106
     1 +TLAMY(3,NFBEAM)*ZB                                                    GEOMA107
     1 +TLAMY(4,NFBEAM)                                                       GEOMA108
      IF(ABS(YLOAD).LT.ABS(YBEAM))GO TO 200                                   GEOMA109
C                                                                             GEOMA110
  150 CONTINUE                                                                GEOMA111
C                                                                             GEOMA112
      CALL ABAMLT (LOADCD,PINTP(M,L,J),TLAMM(1,1,NB),DX,DY,DZ,GAMZ,GAMY)GEOMA113
C                                                                             GEOMA114
  200 CONTINUE                                                                GEOMA115
  300 CONTINUE                                                                GEOMA116
  400 CONTINUE                                                                GEOMA117
C                                                                             GEOMA118
      IF(KPRLDS.EQ.0)GO TO 420                                                GEOMA119
      J1=1                                                                    GEOMA120
      J2=7                                                                    GEOMA121
      I1=NFNLBB(1,L)                                                          GEOMA122
      I2=I1+6                                                                 GEOMA123
  405 IF(J2.GT.NBOXES) J2=NBOXES                                              GEOMA124
      WRITE(NOUT,9)(I,I=I1,I2)                                                GEOMA125
    9 FORMAT (1H0,4X,7I14)                                                    GEOMA126
      DO 410 M=1,NINTLD                                                       GEOMA127
  410 WRITE(NOUT,10) M,(PINTP(M,L,J),J=J1,J2)                                 GEOMA128
   10 FORMAT (1H ,I4,7E14.6)                                                  GEOMA129
      IF(J2.GE.NBOXES)GO TO 420                                               GEOMA130
      J1=J2+1                                                                 GEOMA131
      J2=J1+6                                                                 GEOMA132
      I1=I2+1                                                                 GEOMA133
      I2=I1+6                                                                 GEOMA134
      GO TO 405                                                               GEOMA135
  420 CONTINUE                                                                GEOMA136
C                                                                             GEOMA137
  500 CONTINUE                                                                GEOMA138
C                                                                             GEOMA139
C                                                                             GEOMA140
      RETURN                                                                  GEOMA141
      END                                                                     GEOMA142
```

```
      SUBROUTINE GEOMMB(NMGRP,NFNLMB,NBEAMS,NBCON,NBSEQ,NINTLD,STALDS,
     1 PINTF, PINTL, PINTH, PINTT, PINTA, PINTP,
     2 ELXIO, ELYIO, ELZIO, TLAMY, TLAMM, TLAMV )
C
C     THIS SUBROUTINE CALCS THE INTEGRATED LOADS IN TERMS OF ALL
C     UNIT INERTIAL LOADS AND THEN IN TERMS OF GEN. RESPONSE
C
C
      DIMENSION NFNLMB(3,1)
      DIMENSION NBCON(1)
      DIMENSION NBSEQ(NBEAMS,1)
      DIMENSION STALDS(NINTLD,8)
      DIMENSION PINTF(NINTLD,NMGRP,1)
      DIMENSION PINTL(NINTLD,NMGRP,1)
      DIMENSION PINTH(NINTLD,NMGRP,1)
      DIMENSION PINTT(NINTLD,NMGRP,1)
      DIMENSION PINTA(NINTLD,NMGRP,1)
      DIMENSION PINTP(NINTLD,NMGRP,1)
      DIMENSION ELXIO(1),ELYIO(1),ELZIO(1)
      DIMENSION TLAMY(4,1)
      DIMENSION TLAMM(3,3,1)
      DIMENSION TLAMV(3,3,1)
C
      COMMON /ZZZ/HEDR(49),NOUT,KROW,LINES,IPRNT,NER
      COMMON/XTF/NF(100)
      EQUIVALENCE (NF(24),KPRLDS),   (NF(80),ISECT)
C
C     PINTH,L,F ARE INT.LDS. AT DESIGNATED STA. DUE TO H,L,F FORCES
C     WHERE H,L,F ARE EQUILIBRIUM FORCES IN THE +Z,+Y,+X AAS DIRECTIONS
C     M=INTGD LD. NO.
C     L=MASS GRP NO.
C     J=MASS NO. IN LTH GRP.
C
C
C     LOOP THRU ALL MASS GROUPS
C
      DO 500 L=1,NMGRP
      NMASS=NFNLMB(2,L)-NFNLMB(1,L)+1
      NFBEAM=NFNLMB(3,L)
      KK=NBCON(NFBEAM)
      DO 40  M=1,NINTLD
      DO 40  J=1,NMASS
      PINTH(M,L,J) = 0.0
      PINTL(M,L,J) = 0.0
      PINTF(M,L,J) = 0.0
      PINTT(M,L,J) = 0.0
      PINTA(M,L,J) = 0.0
   40 PINTP(M,L,J) = 0.0
C
      WRITE(NOUT,8)L,NFNLMB(1,L),NFNLMB(2,L)
    8 FORMAT(1H1,20X,*INERTIAL LOAD DATA*/
     1  1H0,20X,*GEOMETRY FOR MASS GROUP*,I4,* MASS*,
```

```
     1  I4,* TO*,I4/1HO,*MASS*,6X,*X*,11X,*Y*,11X,*Z*)                 GEOMMB54
C                                                                       GEOMMB55
      DO 100 J=1,NMASS                                                  GEOMMB56
      LOC=NFNLMB(1,L)+J-1                                               GEOMMB57
      WRITE (NOUT,7)LOC,FLXIO(LOC),ELYIO(LOC) FLZIO(LOC)                GEOMMB58
 100  CONTINUE                                                          GEOMMB59
    7 FORMAT (1H ,I4,3E14.6)                                            GEOMMB60
C                                                                       GEOMMB61
C                                                                       GEOMMB62
C      FIND ALL BEAMS THIS MASS GROUP LOADS UP                          GEOMMB63
C                                                                       GEOMMB64
      DO 400 K=1,KK                                                     GEOMMB65
      NBEAM=NBSFQ(NFBEAM,K)                                             GEOMMB66
C                                                                       GEOMMB67
C     FIND ALL INTEGRATED LOAD STA. THIS MASS GROUP LOADS UP            GEOMMB68
C                                                                       GEOMMB69
      DO 300 M=1,NINTLD                                                 GEOMMB70
      NB=STALDS(M,1)                                                    GEOMMB71
      IF  (NB.NE.NBEAM)  GO TO 300                                      GEOMMB72
      XB=STALDS(M,4)                                                    GEOMMB73
      YB=STALDS(M,5)                                                    GEOMMB74
      ZB=STALDS(M,6)                                                    GEOMMB75
      LOADCD=STALDS(M,2)                                                GEOMMB76
C                                                                       GEOMMB77
C     NOW CALC INT.LOADS FOR ALL UNIT  INERTIA LOADS IN THIS MASS GROUP GEOMMB78
C                                                                       GEOMMB79
      DO 200 J=1,NMASS                                                  GEOMMB80
      LOC=NFNLMB(1,L)+J-1                                               GEOMMB81
      XM=ELXIO(LOC)                                                     GEOMMB82
      YM=ELYIO(LOC)                                                     GEOMMB83
      ZM=ELZIO(LOC)                                                     GEOMMB84
      DX=XM-XB                                                          GEOMMB85
      DY=YM-YB                                                          GEOMMB86
      DZ=ZM-ZB                                                          GEOMMB87
C                                                                       GEOMMB88
C     TEST FOR A MASS IB OF THIS MASS GRP ENTRY BEAM  DON T USE IT      GEOMMB89
C                                                                       GEOMMB90
      IF(NB.NE.NFBEAM)GO TO 150                                         GEOMMB91
      YLOAD=TLAMY(1,NFBEAM)*XM                                          GEOMMB92
     1 +TLAMY(2,NFBEAM)*YM                                              GEOMMB93
     1 +TLAMY(3,NFBEAM)*ZM                                              GEOMMB94
     1 +TLAMY(4,NFBEAM)                                                 GEOMMB95
      YBEAM=TLAMY(1,NFBEAM)*XB                                          GEOMMB96
     1 +TLAMY(2,NFBEAM)*YB                                              GEOMMB97
     1 +TLAMY(3,NFBEAM)*ZB                                              GEOMMB98
     1 +TLAMY(4,NFBEAM)                                                 GEOMMB99
      IF(ABS(YLOAD).LT.ABS(YBEAM))GO TO 200                             GEOMM100
C                                                                       GEOMM101
  150 CONTINUE                                                          GEOMM102
C                                                                       GEOMM103
C                                                                       GEOMM104
C     LOADCD DESIGNATES INTG D LOAD TYPE                                GEOMM105
```

```
C       1=M                                                              GEOMM106
C       2=T                                                              GEOMM107
C       3=Y                                                              GEOMM108
C       4=H                                                              GEOMM109
C       5=L                                                              GEOMM110
C       6=F                                                              GEOMM111
C                                                                        GEOMM112
        CALL MMBMLT (LOADCD, ISECT, TLAMM(1,1,NB), TLAMV(1,1,L), DX,DY,DZ GEOMM113
     1  , PINTF(M,L,J), PINTL(M,L,J), PINTH(M,L,J)                       GEOMM114
     2  , PINTT(M,L,J), PINTA(M,L,J), PINTP(M,L,J) )                     GEOMM115
C                                                                        GEOMM116
  200   CONTINUE                                                         GEOMM117
  300   CONTINUE                                                         GEOMM118
  400   CONTINUE                                                         GEOMM119
C                                                                        GEOMM120
        IF(KPRLDS.EQ.0)GO TO 490                                         GEOMM121
        WRITE(NOUT,6)                                                    GEOMM122
  6     FORMAT(1H0,20X,*INTEGRATED LOADS PER INERTIA LOADS*/1H0,         GEOMM123
     1  20X,*Z LOADS*/1H0,*LOAD*,20X,*MASS NOS.*)                        GEOMM124
    9 FORMAT (1H0,4X,7I14)                                               GEOMM125
   10 FORMAT (1H ,I4,7E14.6)                                             GEOMM126
  11    FORMAT(1H0,20X,*Y LOADS*/1H0,20X,*MASS NOS.*)                    GEOMM127
  12    FORMAT(1H0,20X,*X LOADS*/1H0,20X,*MASS NOS.*)                    GEOMM128
   13 FORMAT (1H0,20X,*Z MOMENTS*/1H0,20X,*MASS NOS.*)                   GEOMM129
   14 FORMAT (1H0,20X,*Y MOMENTS*/1H0,20X,*MASS NOS.*)                   GEOMM130
   15 FORMAT (1H0,20X,*X MOMENTS*/1H0,20X,*MASS NOS.*)                   GEOMM131
C                                                                        GEOMM132
        I0 = NFNUMB(1,L) - 1                                             GEOMM133
C                                                                        GEOMM134
        DO 470 J1=1,NMASS,7                                              GEOMM135
        J2 = J1+6                                                        GEOMM136
        IF (J2.GT.NMASS) J2=NMASS                                        GEOMM137
        I1 = I0+J1                                                       GEOMM138
        I2 = I0+J2                                                       GEOMM139
        WRITE(NOUT,9)(I,I=I1,I2)                                         GEOMM140
        DO 410 M=1,NINTLD                                                GEOMM141
  410   WRITE(NOUT,10)M,(PINTH(M,L,J),J=J1,J2)                           GEOMM142
        WRITE(NOUT,11)                                                   GEOMM143
        WRITE(NOUT,9)(I,I=I1,I2)                                         GEOMM144
        DO 420 M=1,NINTLD                                                GEOMM145
  420   WRITE(NOUT,10)M,(PINTL(M,L,J),J=J1,J2)                           GEOMM146
        WRITE(NOUT,12)                                                   GEOMM147
        WRITE(NOUT,9)(I,I=I1,I2)                                         GEOMM148
        DO 430 M=1,NINTLD                                                GEOMM149
  430   WRITE(NOUT,10)M,(PINTF(M,L,J),J=J1,J2)                           GEOMM150
        WRITE  (NOUT,13)                                                 GEOMM151
        WRITE  (NOUT,9)  (I,I=I1,I2)                                     GEOMM152
        DO 440 M=1,NINTLD                                                GEOMM153
  440   WRITE  (NOUT,10)  M, (FINTP(M,L,J),J=J1,J2)                      GEOMM154
        WRITE  (NOUT,14)                                                 GEOMM155
        WRITE  (NOUT,9)  (I,I=I1,I2)                                     GEOMM156
        DO 450 M=1,NINTLD                                                GEOMM157
```

```
  450 WRITE  (NOUT,10)  M, (PINTA(M,L,J),J=J1,J2)                  GEOMM158
      WRITE  (NOUT,15)                                             GEOMM159
      WRITE  (NOUT,9)  (I,I=I1,I2)                                 GEOMM160
      DO 460 M=1,NINTLD                                            GEOMM161
  460 WRITE  (NOUT,10)  M, (PINTT(M,L,J),J=J1,J2)                  GEOMM162
C                                                                  GEOMM163
  470 CONTINUE                                                     GEOMM164
C                                                                  GEOMM165
  490 CONTINUE                                                     GEOMM166
C                                                                  GEOMM167
  500 CONTINUE                                                     GEOMM168
C                                                                  GEOMM169
C                                                                  GEOMM170
      RETURN                                                       GEOMM171
      END                                                          GEOMM172
```

```
      SUBROUTINE GEOMSB(NSBGRP,NFNLSB,NBEAMS,NBCON,NBSEQ,NINTLD,STALDS,  GEOMSB 2
     1 PINTZ,PINTY,SBXIO ,SBYIO,SBZIO,TLAMY,TLAMM,                       GEOMSB 3
     1 XIS1,ETAS1,ZETAS1,XIS2,ETAS2,ZETAS2)                              GEOMSB 4
C                                                                        GEOMSB 5
C     THIS SUBROUTINE CALCS THE INTEGRATED LOADS IN TERMS OF ALL         GEOMSB 6
C     UNIT S.B.AERO LOADS AND THEN IN TERMS OF GEN. RESPONSE             GEOMSB 7
C                                                                        GEOMSB 8
C                                                                        GEOMSB 9
      DIMENSION NFNLSB(3,1)                                              GEOMSB10
      DIMENSION NBCON(1)                                                 GEOMSB11
      DIMENSION NBSEQ(NBEAMS,1)                                          GEOMSB12
      DIMENSION STALDS(NINTLD,8)                                         GEOMSB13
      DIMENSION PINTZ(NINTLD,NSBGRP,1)                                   GEOMSB14
      DIMENSION PINTY(NINTLD,NSBGRP,1)                                   GEOMSB15
      DIMENSION XIS1(1),XIS2(1)                                          GEOMSB16
      DIMENSION ETAS1(1),ETAS2(1)                                        GEOMSB17
      DIMENSION ZETAS1(1),ZETAS2(1)                                      GEOMSB18
      DIMENSION SBXIO(1),SBYIO(1),SBZIO(1)                               GEOMSB19
      DIMENSION TLAMY(4,1)                                               GEOMSB20
      DIMENSION TLAMM(3,3,1)                                             GEOMSB21
C                                                                        GEOMSB22
      COMMON /ZZZ/HEDR(49),NOUT,KROW,LINES,IPRNT,NER                     GEOMSB23
      COMMON/XTF/NF(100)                                                 GEOMSB24
      EQUIVALENCE (NF(24),KPRLDS)                                        GEOMSB25
C                                                                        GEOMSB26
C     LOOP THRU ALL S.B. GROUPS                                          GEOMSB27
C                                                                        GEOMSB28
      DO 500 L=1,NSBGRP                                                  GEOMSB29
      NAERSB=NFNLSB(2,L)-NFNLSB(1,L)+1                                   GEOMSB30
      NFBEAM=NFNLSB(3,L)                                                 GEOMSB31
      KK=NBCON(NFBEAM)                                                   GEOMSB32
      DO 40 M=1,NINTLD                                                   GEOMSB33
      DO 40 J=1,NAERSB                                                   GEOMSB34
      PINTY(M,L,J) = 0.0                                                 GEOMSB35
   40 PINTZ(M,L,J) = 0.0                                                 GEOMSB36
C                                                                        GEOMSB37
      WRITE(NOUT,8)L,NFNLSB(1,L),NFNLSB(2,L)                             GEOMSB38
    8 FORMAT(1H1,20X,*BODY LOAD DATA*/1H0,20X,                           GEOMSB39
     1  *GEOMETRY FOR BODY GROUP*,I4,* BODIES*,                          GEOMSB40
     1  I4,* TO*,I4/1H0,*BODY*,6X,*X*,11X,*Y*,11X,*Z*)                   GEOMSB41
C                                                                        GEOMSB42
   50 CONTINUE                                                           GEOMSB43
C                                                                        GEOMSB44
C     FIND DIR COS AND S.B. FORCE COORD                                  GEOMSB45
      LOC=NFNLSB(1,L)                                                    GEOMSB46
      GAMY=1.0                                                           GEOMSB47
      IF(XIS2(LOC).LT.XIS1(LOC))GAMY=-1.0                                GEOMSB48
      GAMZ=1.0                                                           GEOMSB49
      DO 100 J=1,NAERSB                                                  GEOMSB50
      LOC=NFNLSB(1,L)+J-1                                                GEOMSB51
      SBXIO(J)=(XIS2(LOC)+XIS1(LOC))/2.                                  GEOMSB52
      SBYIO(J)=(ETAS2(LOC)+ETAS1(LOC))/2.                                GEOMSB53
```

```
      SBZIO(J)=(ZETAS2(LOC)+ZETAS1(LOC))/2.                             GEOMSB54
C                                                                       GEOMSB55
       WRITE(NOUT,7)LOC,SBXIO(J),SBYIO(J),SBZIO(J)                      GEOMSB56
    7 FORMAT (1H ,I4,3E14.6)                                            GEOMSB57
C                                                                       GEOMSB58
  100 CONTINUE                                                          GEOMSB59
C                                                                       GEOMSB60
      IF(KPRLDS.EQ.0)GO TO 105                                          GEOMSB61
      WRITE(NOUT,6)                                                     GEOMSB62
  6   FORMAT(1H0,20X,*INTEGRATED LOAD PER Z BODY LOAD*/                 GEOMSB63
     1  1H0,*LOAD*,20X,*BODY NOS.*)                                     GEOMSB64
C                                                                       GEOMSB65
C                                                                       GEOMSB66
  105  CONTINUE                                                         GEOMSB67
C                                                                       GEOMSB68
C      FIND ALL BEAMS THIS S.B. GROUP LOADS UP                          GEOMSB69
C                                                                       GEOMSB70
      DO 400 K=1,KK                                                     GEOMSB71
      NBEAM=NBSEQ(NFBEAM,K)                                             GEOMSB72
C                                                                       GEOMSB73
C     FIND ALL INTEGRATED LOAD STA. THIS S.B. GROUP LOADS UP            GEOMSB74
C                                                                       GEOMSB75
      DO 300 M=1,NINTLD                                                 GEOMSB76
      NB=STALDS(M,1)                                                    GEOMSB77
      IF  (NB.NE.NBEAM) GO TO 300                                       GEOMSB78
C                                                                       GEOMSB79
      XB=STALDS(M,4)                                                    GEOMSB80
      YB=STALDS(M,5)                                                    GEOMSB81
      ZB=STALDS(M,6)                                                    GEOMSB82
      LOADCD=STALDS(M,2)                                                GEOMSB83
C                                                                       GEOMSB84
C     NOW CALC INT.LOADS FOR ALL UNIT S.B.AERO LOADS IN THIS S.B. GROUP GEOMSB85
C                                                                       GEOMSB86
      DO 200 J=1,NAERSB                                                 GEOMSB87
      XM=SBXIO(J)                                                       GEOMSB88
      YM=SBYIO(J)                                                       GEOMSB89
      ZM=SBZIO(J)                                                       GEOMSB90
      DX=XM-XB                                                          GEOMSB91
      DY=YM-YB                                                          GEOMSB92
       DZ=ZM-ZB                                                         GEOMSB93
C                                                                       GEOMSB94
C     TEST FOR A S.B. IB OF THIS S.B. GRP ENTRY BEAM  DON T USE IT      GEOMSB95
C                                                                       GEOMSB96
      IF(NB.NE.NFBEAM)GO TO 150                                         GEOMSB97
       YLOAD=TLAMY(1,NFBEAM)*XM                                         GEOMSB98
     1 +TLAMY(2,NFBEAM)*YM                                              GEOMSB99
     1 +TLAMY(3,NFBEAM)*ZM                                              GEOMS100
     1 +TLAMY(4,NFBEAM)                                                 GEOMS101
       YBEAM=TLAMY(1,NFBEAM)*XB                                         GEOMS102
     1 +TLAMY(2,NFBEAM)*YB                                              GEOMS103
     1 +TLAMY(3,NFBEAM)*ZB                                              GEOMS104
     1 +TLAMY(4,NFBEAM)                                                 GEOMS105
```

```
      IF(ABS(YLOAD).LT.ABS(YBEAM))GO TO 200                            GEOMS106
C                                                                      GEOMS107
  150 CONTINUE                                                         GEOMS108
      CALL SBAMLT (LOADCD,PINTZ(M,L,J),PINTY(M,L,J),TLAMM(1,1,NB)      GEOMS109
     1 , DX,DY,DZ,GAMZ,GAMY)                                           GEOMS110
C                                                                      GEOMS111
  200 CONTINUE                                                         GEOMS112
  300 CONTINUE                                                         GEOMS113
  400 CONTINUE                                                         GEOMS114
C                                                                      GEOMS115
      IF(KPRLDS.EQ.0)GO TO 440                                         GEOMS116
      J1=1                                                             GEOMS117
      J2=7                                                             GEOMS118
      I1=NFNLSB(1,L)                                                   GEOMS119
      I2=I1+6                                                          GEOMS120
  405 IF(J2.GT.NAERSB) J2=NAERSB                                       GEOMS121
      WRITE(NOUT,9)(I,I=I1,I2)                                         GEOMS122
    9 FORMAT (1H ,4X,7I14)                                             GEOMS123
      DO 410 M=1,NINTLD                                                GEOMS124
  410 WRITE(NOUT,10) M,(PINTZ(M,L,J),J=J1,J2)                          GEOMS125
   10 FORMAT (1P ,I4,7E14.6)                                           GEOMS126
      IF(J2.GE.NAERSB)GO TO 420                                        GEOMS127
      J1=J2+1                                                          GEOMS128
      J2=J1+6                                                          GEOMS129
      I1=I2+1                                                          GEOMS130
      I2=I1+6                                                          GEOMS131
      GO TO 405                                                        GEOMS132
  420 CONTINUE                                                         GEOMS133
      WRITE(NOUT,5)                                                    GEOMS134
    5 FORMAT(1H0,20X,*INTEGRATED LOADS PER Y BODY LOAD*/               GEOMS135
     1 1H0,*LOAD*,20X,*BODY NOS.*)                                     GEOMS136
      J1=1                                                             GEOMS137
      J2=7                                                             GEOMS138
      I1=NFNLSB(1,L)                                                   GEOMS139
      I2=I1+6                                                          GEOMS140
  425 IF(J2.GT.NAERSB) J2=NAERSB                                       GEOMS141
      WRITE(NOUT,9)(I,I=I1,I2)                                         GEOMS142
      DO 430 M=1,NINTLD                                                GEOMS143
  430 WRITE(NOUT,10) M,(PINTY(M,L,J),J=J1,J2)                          GEOMS144
      IF(J2.GE.NAERSB)GO TO 440                                        GEOMS145
      J1=J2+1                                                          GEOMS146
      J2=J1+6                                                          GEOMS147
      I1=I2+1                                                          GEOMS148
      I2=I1+6                                                          GEOMS149
      GO TO 425                                                        GEOMS150
C                                                                      GEOMS151
  440 CONTINUE                                                         GEOMS152
C                                                                      GEOMS153
  500 CONTINUE                                                         GEOMS154
C                                                                      GEOMS155
      RETURN                                                           GEOMS156
      END                                                              GEOMS157
```

```
      SUBROUTINE INPUTL (NTI,BEAMGM,NBEAMS,STALDS,NINTLD,STRESS,NSTRSS
     1 , RFNLMB,NMGRP,RFNLAB,NABGRP,RFNLSB,NSBGRP,NMASS,NBOXES,NAERSB
     2 , TLAMV,NENGS, ENGM)
C
      COMMON/ZZZ/HEDR(48),NIN,NOUT,KROW,LINES,IPRNT,NER
C
      DIMENSION BEAMGM(NBEAMS,1), STALDS(NINTLD,1), STRESS(NSTRSS,1)
      DIMENSION RFNLMB(3,1), RFNLAB(3,1), RFNLSB(3,1)
      DIMENSION TLAMV(3,3,1)
      DIMENSION  ENGM(1)
C
  310 FORMAT (6F12.0)
C
      DO 100 I=1,NBEAMS
      READ (NTI,310) (BEAMGM(I,J),J=1,3)
      READ (NTI,310) (BEAMGM(I,J),J=4,7)
  100 CONTINUE
      CALL HEADNG
      WRITE(NOUT,1)
    1 FORMAT(1H0,20X, UNIT LOAD MODULE ANALYSIS )
      CALL PRNT (BEAMGM,NBEAMS,7,NBEAMS,1,6HBEAMGM,2)
C
      DO 120 I=1,NINTLD
      READ (NTI,310) (STALDS(I,J),J=1,3)
      READ (NTI,310) (STALDS(I,J),J=4,8)
  120 CONTINUE
      CALL PRNT (STALDS,NINTLD,8,NINTLD,1,6HSTALDS,2)
C
      IF (NSTRSS.EQ.0) GO TO 150
      DO 140 I=1,NSTRSS
      READ(NTI,310)(STRESS(I,J),J=1,NINTLD)
  140 CONTINUE
      CALL PRNT (STRESS,NSTRSS,NINTLD,NSTRSS,1,6HSTRESS,2)
C
  150 NMASS = 0
      DO 200 J=1,NMGRP
      READ (NTI,310) (RFNLMB(I,J),I=1,3)
      ITEST = RFNLMB(2,J) - RFNLMB(1,J) + 1
      NMASS = MAX0(NMASS,ITEST)
  200 CONTINUE
      CALL PRNT (RFNLMB,3,NMGRP,3,1,6HRFNLMB,2)
C
      DO 205 N=1,NMGRP
      DO 204 I=1,3
  204 READ(NTI,310)(TLAMV(I,J,N),J=1,3)
      WRITE(NOUT,202)N
  202 FORMAT(1H0, MASS GROUP = ,I4)
  205 CALL PRNT (TLAMV(1,1,N),3,3,3,1,4HLAMV,1)
C
      NBOXES = 0
      DO 210 J=1,NABGRP
      READ (NTI,310) (RFNLAB(I,J),I=1,3)
```

```
      ITEST = RENLAB(2,J) - RENLAB(1,J) + 1
      NBOXES = MAXO(NBOXES,ITEST)
  210 CONTINUE
      CALL PRNT (RENLAB,3,NABGRP,3,1,6HNENLAB,2)
C
      IF (NSBGRP.EQ.0) GO TO 250
      NAERSB = 0
      DO 220 J=1,NSBGRP
      READ (NTI,310) (RENLSB(I,J),I=1,3)
      ITEST = RENLSB(2,J) - RENLSB(1,J) + 1
      NAERSB = MAXO(NAERSB,ITEST)
  220 CONTINUE
      CALL PRNT (RENLSB,3,NSBGRP,3,1,6HNENLSB,2)
C
  250 IF(NENGS.EQ.0)GO TO 260
      II=-1
      DO 255 J=1,NENGS
      II=II+2
      READ(NTI,310) ENGM(II), ENGM(II+1)
  255 CONTINUE
      CALL PRNT(ENGM,NENGS,2,NENGS,1,4HENGM,1)
  260 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE MMBMLT (LOADCD, ISECT, TLAMM, TLAMV, DX, DY, DZ          MMBMLT 2
     1 , PINTF, PINTL, PINTH, PINTT, PINTA, PINTP )                      MMBMLT 3
C                                                                        MMBMLT 4
C                                                                        MMBMLT 5
      DIMENSION TLAMM(3,3),TLAMV(3,3)                                    MMBMLT 6
C                                                                        MMBMLT 7
C                                                                        MMBMLT 8
      GO TO (501,502,503,504,505,506),LOADCD                             MMBMLT 9
C                                                                        MMBMLT10
  501 AF=TLAMM(1,2)*DZ-TLAMM(1,3)*DY                                     MMBMLT11
      AL=-TLAMM(1,1)*DZ+TLAMM(1,3)*DX                                    MMBMLT12
      AH=TLAMM(1,1)*DY-TLAMM(1,2)*DX                                     MMBMLT13
      IF (ISECT.EQ.0) GO TO 510                                          MMBMLT14
      AT = TLAMM(1,1)                                                    MMBMLT15
      AA = TLAMM(1,2)                                                    MMBMLT16
      AP = TLAMM(1,3)                                                    MMBMLT17
      GO TO 510                                                          MMBMLT18
C                                                                        MMBMLT19
  502 AF=TLAMM(2,2)*DZ-TLAMM(2,3)*DY                                     MMBMLT20
      AL=-TLAMM(2,1)*DZ+TLAMM(2,3)*DX                                    MMBMLT21
      AH=TLAMM(2,1)*DY-TLAMM(2,2)*DX                                     MMBMLT22
      IF (ISECT.EQ.0) GO TO 510                                          MMBMLT23
      AT = TLAMM(2,1)                                                    MMBMLT24
      AA = TLAMM(2,2)                                                    MMBMLT25
      AP = TLAMM(2,3)                                                    MMBMLT26
      GO TO 510                                                          MMBMLT27
C                                                                        MMBMLT28
  503 AF=TLAMM(3,2)*DZ-TLAMM(3,3)*DY                                     MMBMLT29
      AL=-TLAMM(3,1)*DZ-TLAMM(3,3)*DX                                    MMBMLT30
      AH=TLAMM(3,1)*DY-TLAMM(3,2)*DX                                     MMBMLT31
      IF (ISECT.EQ.0) GO TO 510                                          MMBMLT32
      AT = TLAMM(3,1)                                                    MMBMLT33
      AA = TLAMM(3,2)                                                    MMBMLT34
      AP = TLAMM(3,3)                                                    MMBMLT35
      GO TO 510                                                          MMBMLT36
C                                                                        MMBMLT37
  504 AF=TLAMM(3,1)                                                      MMBMLT38
      AL=TLAMM(3,2)                                                      MMBMLT39
      AH=TLAMM(3,3)                                                      MMBMLT40
      GO TO 510                                                          MMBMLT41
C                                                                        MMBMLT42
  505 AF=TLAMM(2,1)                                                      MMBMLT43
      AL=TLAMM(2,2)                                                      MMBMLT44
      AH=TLAMM(2,3)                                                      MMBMLT45
      GO TO 510                                                          MMBMLT46
C                                                                        MMBMLT47
  506 AF =TLAMM(1,1)                                                     MMBMLT48
      AL=TLAMM(1,2)                                                      MMBMLT49
      AH=TLAMM(1,3)                                                      MMBMLT50
C                                                                        MMBMLT51
C                                                                        MMBMLT52
  510 CONTINUE                                                           MMBMLT53
```

```
C     NOW ROTATE TO INERTIAL LOAD COORDS                                MMBMLT54
C                                                                       MMBMLT55
      PINTF=AF*TLAMV(1,1)+AL*TLAMV(2,1)+AH*TLAMV(3,1)                   MMBMLT56
      PINTL=AF*TLAMV(1,2)+AL*TLAMV(2,2)+AH*TLAMV(3,2)                   MMBMLT57
      PINTH=AF*TLAMV(1,3)+AL*TLAMV(2,3)+AH*TLAMV(3,3)                   MMBMLT58
      IF (ISECT.EQ.0) GO TO 600                                         MMBMLT59
      PINTT = 0.0                                                       MMBMLT60
      PINTA = 0.0                                                       MMBMLT61
      PINTP = 0.0                                                       MMBMLT62
      IF (LOADCD.GT.3) GO TO 600                                        MMBMLT63
      PINTT = AT*TLAMV(1,1) + AA*TLAMV(2,1) + AP*TLAMV(3,1)             MMBMLT64
      PINTA = AT*TLAMV(1,2) + AA*TLAMV(2,2) + AP*TLAMV(3,2)             MMBMLT65
      PINTP = AT*TLAMV(1,3) + AA*TLAMV(2,3) + AP*TLAMV(3,3)             MMBMLT66
C                                                                       MMBMLT67
C                                                                       MMBMLT68
  600 RETURN                                                            MMBMLT69
      END                                                               MMBMLT70
```

```
      SUBROUTINE MSSPHI (NMGRP,NFNLMB,NINTLD
     1 , PINTF , PINTL, PINTH, PINTT, PINTA, PINTP
     2 , NDOF,NMS,FMPHI,NM,PIQ,SYMCOD,ASMCOD,NSYM,NASYM )
C
      COMMON/XTF/NF(100)
      EQUIVALENCE (NF(24),KPRCHK)
C
      DIMENSION PIQ(NINTLD,1)
      DIMENSION PINTF(NINTLD,NMGRP,1)
      DIMENSION PINTL(NINTLD,NMGRP,1)
      DIMENSION PINTH(NINTLD,NMGRP,1)
      DIMENSION PINTT(NINTLD,NMGRP,1)
      DIMENSION PINTA(NINTLD,NMGRP,1)
      DIMENSION PINTP(NINTLD,NMGRP,1)
      DIMENSION FMPHI(NDOF,NMS,1)
      DIMENSION NFNLMB(3,1)
      DIMENSION SYMCOD(1),ASMCOD(1)
C
C
C     CALC INTGD INERTIAL LOADS DUE TO UNIT MODAL AMPLITUDES
C
      DO 100 M=1,NINTLD
      DO 100 N=1,NM
      PIQ(M,N)=0.
      DO 100 L=1,NMGRP
      NMASS=NFNLMB(2,L)-NFNLMB(1,L)+1
      DO 90 J=1,NMASS
      LOC=NFNLMB(1,L)+J-1
      PIQ(M,N)=PIQ(M,N)+PINTF(M,L,J)*FMPHI(1,LOC,N)
     1  +PINTL(M,L,J)*FMPHI(2,LOC,N)
     1  +PINTH(M,L,J)*FMPHI(3,LOC,N)
      IF (NDOF.LT.4) GO TO 90
      PIQ(M,N) = PIQ(M,N) + PINTT(M,L,J) * FMPHI(4,LOC,N)
      IF (NDOF.LT.5) GO TO 90
      PIQ(M,N) = PIQ(M,N) + PINTA(M,L,J) * FMPHI(5,LOC,N)
      IF (NDOF.LT.6) GO TO 90
      PIQ(M,N) = PIQ(M,N) + PINTP(M,L,J) * FMPHI(6,LOC,N)
   90 CONTINUE
  100 CONTINUE
C
C     NOW CHECK FOR SYM,ASYM CORRECTIONS ON C/L LOADS
C
      DO 300 M=1,NINTLD
      IF(SYMCOD(M).EQ.1.AND.ASMCOD(M).EQ.1)GO TO 300
      DO 200 N=1,NSYM
  200 PIQ(M,N)=PIQ(M,N)*SYMCOD(M)
      DO 250 N=1,NASYM
      NN=N+NSYM
  250 PIQ(M,NN)=PIQ(M,NN)*ASMCOD(M)
  300 CONTINUE
C
C
```

```
      IF(KPRC+K.NE.0)CALL PRNT(PIQ,NINTLD,NM,NINTLD,1,4H PIQ ,1)         MSSPHI54
C                                                                        MSSPHI55
C                                                                        MSSPHI56
      RETURN                                                             MSSPHI57
      END                                                                MSSPHI58
```

```
      SUBROUTINE RTOI (R,I,N)                                           RTOI   2
      DIMENSION R(1),I(1)                                               RTOI   3
      DO 100 M=1,N                                                      RTOI   4
  100 I(M) = R(M)                                                       RTOI   5
      RETURN                                                            RTOI   6
      END                                                               RTOI   7
```

```
      SUBROUTINE SBAMLT(LOADCD,PINTZ,PINTY,TLAMM,DX,DY,DZ,GAMZ,GAMY)
C
      DIMENSION TLAMM(3,3)
C
      GO TO (501,502,503,504,505,506),LOADCD
C
  501 AL=-TLAMM(1,1)*DZ+TLAMM(1,3)*DX
      AH=TLAMM(1,1)*DY-TLAMM(1,2)*DX
      GO TO 510
C
  502 AL=-TLAMM(2,1)*DZ+TLAMM(2,3)*DX
      AH=TLAMM(2,1)*DY-TLAMM(2,2)*DX
      GO TO 510
C
  503 AL=-TLAMM(3,1)*DZ+TLAMM(3,3)*DX
      AH=TLAMM(3,1)*DY-TLAMM(3,2)*DX
      GO TO 510
C
  504 AL=TLAMM(3,2)
      AH=TLAMM(3,3)
      GO TO 510
C
  505 AL=TLAMM(2,2)
      AH=TLAMM(2,3)
      GO TO 510
C
  506 AL=TLAMM(1,2)
      AH=TLAMM(1,3)
C
C
  510 CONTINUE
C
C     NEW ROTATE TO AERO LOAD SYSTEM
C
      PINTZ=AH*GAMZ
      PINTY=AL*GAMY
C
C
      RETURN
      END
```

```
      SUBROUTINE SYMSOL(NINTLD,SYMCOD,ASMCOD,STALDS,TLAMM)
C
      DIMENSION STALDS(NINTLD,1),TLAMM(3,3,1),SYMCOD(1),ASMCOD(1)
C
      COMMON/XTF/XF(100)
      EQUIVALENCE (XF(24),KPRLDS)
C
      DATA EPL/0.01/
C
C     NOW FIND INTGD LOADS ON C/L, SET UP MATRIX MOD FOR EACH
C
C     CHECK FOR C/L LOADS,IF NOT ON C/L THEN IT IS NOT MODED
C
      DO 500 I=1,NINTLD
      SYMCOD(I)=1.0
      ASMCOD(I)=1.0
      Y=STALDS(I,5)
      IF(ABS(Y).GT.EPL)GO TO 500
C
C     FOUND C/L LOAD FIND DIR COS, IF SYM LOAD SET ASMCOD=0,
C     IF ASYM LOAD SET SYMCOD=0, IF MIXED SET APPROP. VAL IN ARRAYS
C
      N=STALDS(I,1)
      LOADCD=STALDS(I,2)
      IF(LOADCD.EQ.1.OR.LOADCD.EQ.6)L=1
      IF(LOADCD.EQ.2.OR.LOADCD.EQ.5)L=2
      IF(LOADCD.EQ.3.OR.LOADCD.EQ.4)L=3
C
      XR=TLAMM(L,1,N)
      YR=TLAMM(L,2,N)
      ZR=TLAMM(L,3,N)
C
      IF(LOADCD.GT.3)GO TO 200
C     MOMENT CHECK
C
C     ASYM MOMENT ONLY
      IF(ABS(YR).LT.EPL)GO TO 410
C     SYM MOMENT ONLY
      IF(ABS(ZR).LT.EPL.AND.ABS(XR).LT.EPL)GO TO 400
C     MIXED MOMENT, GET SYM AND ASYM COMPONENTS
C
      SYMCOD(I)=ABS(YR)
      ASMCOD(I)=SQRT(1.-YR**2)
      GO TO 500
C
  200 CONTINUE
C     SHEAR CHECK
C
C     SYM LOAD ONLY
      IF(ABS(YR).LT.EPL)GO TO 400
C     ASYM LOAD ONLY
      IF(ABS(XR).LT.EPL.AND.ABS(ZR).LT.EPL)GO TO 410
```

```
C       MIXED LOAD GET SYM AND ASYM COMPONENTS                              SYMSOL54
C                                                                           SYMSOL55
        ASMCOD(I)=ABS(YR)                                                   SYMSOL56
        SYMCOD(I)=SQRT(1.-YR**2)                                            SYMSOL57
        GO TO 500                                                           SYMSOL58
C                                                                           SYMSOL59
  400 CONTINUE                                                              SYMSOL60
        ASMCOD(I)=0.                                                        SYMSOL61
        GO TO 500                                                           SYMSOL62
  410 CONTINUE                                                              SYMSOL63
        SYMCOD(I)=0.0                                                       SYMSOL64
C                                                                           SYMSOL65
  500 CONTINUE                                                              SYMSOL66
C                                                                           SYMSOL67
        IF(KPRLDS.EQ.0)GO TO 600                                            SYMSOL68
C                                                                           SYMSOL69
        CALL HEADNG                                                         SYMSOL70
        CALL PRNT(SYMCOD,NINTLD,1,NINTLD,1,8H SYMCOD  ,2)                   SYMSOL71
        CALL PRNT(ASMCOD,NINTLD,1,NINTLD,1,8H ASMCOD  ,2)                   SYMSOL72
  600 CONTINUE                                                              SYMSOL73
C                                                                           SYMSOL74
        RETURN                                                              SYMSOL75
        END                                                                 SYMSOL76
```

```
      SUBROUTINE THRUST(
     1 NENGS,NMS,NMGRP,NINTLD,NSYM,
     1 ENGM,ELXIO,FLYIO,ELZIO,PHIX,PHIY,PHIZ,
     1 PINTF,PINTL,PINTH,NFNLMB,TLAMV,
     1 THRLOD,THRGNF)
C
      COMMON/ZZZ/CASE(48),NIN,NOUT,KROW,LINES,IPRNT,NER
C
      COMMON/XTF/NF(100)
      EQUIVALENCE (NF(24),KPRLDS)
C
C     FORM INTEGRATED LOADS DUE TO UNIT THRUST (THRLOD)
C     FORM GENLZD FORCE DUE TO UNIT THRUST SYM ONLY (THRGNF)
C     THRUST ACTS AT FWD MASS OF TWO DEFINED FOR DIRECTION
C
      DIMENSION ENGM(1),ELXIO(1),FLYIO(1),ELZIO(1)
      DIMENSION PHIX(NMS,1),PHIY(NMS,1),PHIZ(NMS,1)
      DIMENSION PINTF(NINTLD,NMGRP,1),PINTL(NINTLD,NMGRP,1),
     1 PINTH(NINTLD,NMGRP,1)
      DIMENSION NFNLMB(3,1),TLAMV(3,3,1)
      DIMENSION THRLOD(NINTLD,NENGS),THRGNF(NSYM,NENGS)
C
C
    1 FORMAT(1H0, OH OH THIS THRUST MASS NOT DEFINED IN NFNLMB,MSNO ,I4)
    2 FORMAT(1H0, OH OH SOMETHING SCREWED UP, LL= ,I4,  JJ= ,I4, MS ,I4)
    3 FORMAT(1H0,20X, INTEGRATED LOADS DUE TO ENGINE NO.  ,I4)
    4 FORMAT(1H0,20X, GENLZD FORCE DUE TO ENGINE NO. ,I4)
    5 FORMAT(20X, THRUST LOADS FROM UNIT THRUST )
    6 FORMAT(1H0,20X, ENGINE THRUST MATRICES GENERATED FOR ,I4,  ENGS )
C
C
      IF(KPRLDS.NE.0)CALL HEADNG
      WRITE(NOUT,6)NENGS
      IF(KPRLDS.NE.0)WRITE(NOUT,5)
       IENG=0
      DO 500 I=1,NENGS,2
C
      IENG=IENG+1
C
      DO 100 K=1,NINTLD
  100 THRLOD(K,IENG)=0.
       DO 110 K=1,NSYM
  110 THRGNF(K,IENG)=0.
C
      NFM=ENGM(I)
      NLM=ENGM(I+1)
C
C     NOW FIND DIR COS OF THRUST
C
      X1=ELXIO(NFM)
      X2=ELXIO(NLM)
      Y1=ELYIO(NFM)
```

```
      Y2=ELYIO(NLM)                                                    THRUST54
      Z1=ELZIO(NFM)                                                    THRUST55
      Z2=ELZIO(NLM)                                                    THRUST56
      DELX=X1-X2                                                       THRUST57
      DELY=Y1-Y2                                                       THRUST58
      DELZ=Z1-Z2                                                       THRUST59
      D=SQRT(DELX**2+DELY**2+DELZ**2)                                  THRUST60
C                                                                      THRUST61
C     TEST THIS ENGINE - IF ON FUSELAGE CL,HALVE THE THRUST            THRUST62
      IF(Y1.EQ.0)D=2.*D                                                THRUST63
C                                                                      THRUST64
      GAMX=DELX/D                                                      THRUST65
      GAMY=DELY/D                                                      THRUST66
      GAMZ=DELZ/D                                                      THRUST67
C                                                                      THRUST68
C     NOW GET THRUST LOAD ALIGNED WITH FWD MASS DOF                    THRUST69
C                                                                      THRUST70
      DO 150 L=1,NMGRP                                                 THRUST71
      LL=L                                                             THRUST72
      NMASS=NFNLMB(2,L)-NFNLMB(1,L)+1                                  THRUST73
      IF(NFM.GE.NFNLMB(1,L).AND.NFM.LE.NFNLMB(2,L))GO TO 160           THRUST74
  150 CONTINUE                                                         THRUST75
      WRITE(NOUT,1)NFM                                                 THRUST76
      GO TO 500                                                        THRUST77
  160 DO 170 J=1,NMASS                                                 THRUST78
      LOC=NFNLMB(1,LL)+J-1                                             THRUST79
      JJ=J                                                             THRUST80
      IF(LOC.EQ.NFM)GO TO 180                                          THRUST81
  170 CONTINUE                                                         THRUST82
      WRITE(NOUT,2)LL,JJ,NFM                                           THRUST83
      GO TO 500                                                        THRUST84
C                                                                      THRUST85
  180 CONTINUE                                                         THRUST86
C     HEY WE FOUND EVERYTHING                                          THRUST87
C     NOW CALC COMPONENTS OF FORCE ALONG INERTIAL DOFS                 THRUST88
C                                                                      THRUST89
      FX=GAMX*TLAMV(1,1,LL)+GAMY*TLAMV(1,2,LL)+GAMZ*TLAMV(1,3,LL)     THRUST90
      FY=GAMX*TLAMV(2,1,LL)+GAMY*TLAMV(2,2,LL)+GAMZ*TLAMV(2,3,LL)     THRUST91
      FZ=GAMX*TLAMV(3,1,LL)+GAMY*TLAMV(3,2,LL)+GAMZ*TLAMV(3,3,LL)     THRUST92
C                                                                      THRUST93
C     NOW CALC INTEGRATED LOADS FROM THIS UNIT THRUST                  THRUST94
C                                                                      THRUST95
      DO 200 K=1,NINTLD                                                THRUST96
  200 THRLOD(K,IENG)=                                                  THRUST97
     1 PINTF(K,LL,JJ)*FX+PINTL(K,LL,JJ)*FY+PINTH(K,LL,JJ)*FZ          THRUST98
C                                                                      THRUST99
C     NOW CALC GENLZD SYM FORCES FROM THIS UNIT THRUST                 THRUS100
C                                                                      THRUS101
      DO 210 K=1,NSYM                                                  THRUS102
  210 THRGNF(K,IENG)=                                                  THRUS103
     1 PHIX(NFM,K)*FX+PHIY(NFM,K)*FY+PHIZ(NFM,K)*FZ                   THRUS104
C                                                                      THRUS105
```

```
C                                                                       THRUS106
      IF(KPRLDS.EQ.0)GO TO 500                                          THRUS107
      WRITE(NOUT,3)IENG                                                 THRUS108
      CALL  PRNT(THRLOD(1,IENG),NINTLD,1,NINTLD,1,4H    ,1)             THRUS109
      WRITE(NOUT,4)IENG                                                 THRUS110
      CALL  PRNT(THRGNF(1,IENG),NSYM,1,NSYM,1,4H    ,1)                 THRUS111
C                                                                       THRUS112
  500 CONTINUF                                                          THRUS113
C                                                                       THRUS114
      RETURN                                                            THRUS115
      END                                                               THRUS116
```

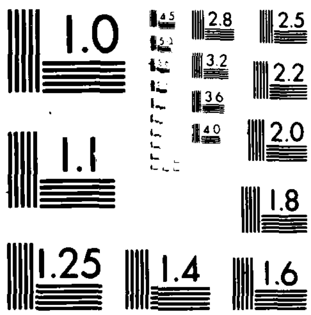
```
      SUBROUTINE ACS(NTI,
     1 MXBLK,MXOBLK,PHI,TMSS,TMSA,TFCS,TFCA,PHISS,PHISA)
C
      COMMON NAA,A(1)
C
      COMMON/ZZZ/HEDR(48),NIN,NOUT,KROW,LINFS
C
      COMMON/XTF/NF(100)
C
      EQUIVALENCE (NF(5),NMS),(NF(6),NSYM),(NF(7),NASYM)
     1 , (NF(8),NOMOD),(NF(70),MXORD),(NF(75),NTFS),(NF(76),NTFA)
C
      DIMENSION PHI(1)
      DIMENSION TMSS(1),TMSA(1),TFCS(1),TFCA(1),PHISS(1),PHISA(1)
C
  999 FORMAT(1H0,20X,I10,  WORDS OF CORE RQD FOR STEP +++ACS STUP+++ )
C
      NTFM=MAXO(NTFS,NTFA)
C
      L1=NAA
      L2=L1+NTFM*MXBLK*MXOBLK
      L3=L2+NTFM*MXBLK*MXOBLK
      L4=L3+2*MXORD
      L5=L4+2*MXORD
      L6=L5+MXBLK*MXOBLK
      L7 = L6 + MXBLK*MXOBLK
C
      WRITE(NOUT,999)L7
C
      CALL ACSMOD(
     1 NTI,NTFM,NTFS,NTFA,NSYM,NASYM,NOMOD,MXORD,MXBLK,MXOBLK,NMS,
     1 PHI,PHISS,PHISA,TMSS,TMSA,TFCS,TFCA,
     1 A(L1),A(L2),A(L3),A(L4),A(L5),A(L6))
C
C
      RETURN
      END
```

```
      SUBROUTINE ACSMOD(NTI,
     1 NTFM,NTFS,NTFA,NSYM,NASYM,NOMOD,MXORD,MXBLK,MXOBLK,NMS,
     1 PHI,PHISS,PHISA,TMSS,TMSA,TFCS,TFCA,
     1 ANUM,ADEN,ATFN,ATFD,ANUMT,ADENT)
C
      COMMON/ZZZ/CASE(48),NIN,NOUT,KROW,LINES,IPRNT,NER
C
      COMMON/XTF/NF(100)
      EQUIVALENCE (NF(81),NDOF)
      EQUIVALENCE (NF(71),MXORSN),(NF(72),MXORSD)
     1 , (NF(73),MXORAN),(NF(74),MXORAD)
C
      DIMENSION TMSS(NTFS,4),TFCS(2,NTFS,MXORD),PHISS(NTFS,NSYM)
      DIMENSION TMSA(NTFA,4),TFCA(2,NTFA,MXORD),PHISA(NTFA,NASYM)
      DIMENSION PHI(1)
      DIMENSION ANUM(NTFM,MXBLK,MXOBLK),ADEN(NTFM,MXBLK,MXOBLK)
      DIMENSION ATFN(2,MXORD),ATFD(2,MXORD)
      DIMENSION ANUMT(MXBLK,MXOBLK),ADENT(MXBLK,MXOBLK)
C
    1 FORMAT(1H0,20X, SYMMETRIC TRANSFER FUNCTION DATA )
    2 FORMAT(1H0,20X, ANTISYMMETRIC TRANSFER FUNCTION DATA )
   20 FORMAT(4F12.0)
C
C
      CALL HEADNG
      WRITE(NOUT,1)
C
      DO 100 I=1,NTFS
  100 READ(NTI,20)(TMSS(I,J),J=1,4)
C
      CALL AFCS (NTFS,TMSS,PHISS,PHI,NSYM,NDOF,NMS,NOMOD,0)
C
      CALL TFBLIN(NTI,NTFS,MXBLK,MXOBLK,ANUM,ADEN,0,1.0,1)
C     NOTE    TF BLOCKS ARE READ IN TFBLIN
C
      CALL TFBLOK(NTFS,MXBLK,MXOBLK,TFCS,ANUM,ADEN,ATFN,ATFD,
     1 ANUMT,ADENT,MXORSN,MXORSD,MXORD)
C
      CALL TRFVAL(NTFS,TFCS,1.0,MXORSN,MXORSD)
C
C
      CALL HEADNG
      WRITE(NOUT,2)
C
      DO 200 I=1,NTFA
  200  READ(NTI,20)(TMSA(I,J),J=1,4)
C
      CALL AFCS (NTFA,TMSA,PHISA,PHI,NASYM,NDOF,NMS,NOMOD,NSYM)
C
      CALL TFBLIN(NTI,NTFA,MXBLK,MXOBLK,ANUM,ADEN,0,1.0,1)
C     NOTE    TF BLOCKS ARE READ IN TFBLIN
C
```

```
      CALL TFBLOK(NTFA,MXBLK,MXOBLK,TFCA,ANUM,ADEN,ATFN,ATFD,          ACSMOD54
     1 ANUMT,ADENT,MXORAN,MXORAD,MXORD)                                ACSMOD55
C                                                                      ACSMOD56
      CALL TRFVAL(NTFA,TFCA,1.0,MXORAN,MXORAD)                         ACSMOD57
C                                                                      ACSMOD58
C                                                                      ACSMOD59
C     NOW SAVE THE FOLLOWING DATA                                      ACSMOD60
C                                                                      ACSMOD61
C     VARIABLES   MXORD,MXORSN,MXORSD,MXORAN,MXORAD,NTFS,NTFA          ACSMOD62
C     ARRAYS  TFCS,TMSS,PHISS,TFCA,TMSA,PHISA                          ACSMOD63
C                                                                      ACSMOD64
C                                                                      ACSMOD65
      RETURN                                                           ACSMOD66
      END                                                              ACSMOD67
```

AD-A106 520 DOUGLAS AIRCRAFT CO. LONG BEACH CA F/G 18/3
NUCLEAR BLAST RESPONSE COMPUTER PROGRAM. VOLUME III. PROGRAM LI--ETC(U)
AUG 81 J A MCGREW, H H CROXEN, T P KALMAN DNA001-75-C-0216
UNCLASSIFIED AFWL-TR-81-32-VOL-3 NL

4 OF 5
AD
A106520



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
      SUBROUTINE AFCS(
     1 NTF,TMS,PHIS,PHI,NM,NDOF,NMS,NOMOD,NO)
C
C
      COMMON/ZZZ/CASE(48),NIN,NOUT,KROW,LINES,IPRNT,NFR
C
      DIMENSION TMS(NTF,4)
      DIMENSION PHIS(NTF,NM), PHI(NMS,NOMOD,1)
C
   40 FORMAT(1H ,3(I4,2X),E13.6,2X,I4)
   60 FORMAT(1H0,20X, ACTIVE SYSTEM COUPLING /1H0,
     1  TFNO  MSNO NCODE      DIR.COS.      MODE )
C
C     NTFN=TF NO.
C     MSNO= MASS NO
C     NCODE=DIRECTION CODE   1=X,2=Y,3=Z
C     DIRCOS=SCALAR MULT FOR DOF
C      NO IS NO. OF MODES PRECEDING SET,0 FOR SYM,NSYM FOR ASYM LOOPS
C       NMS=NO. MASS POINTS
C      NM=NO. MODES(NSYM OR NASYM)
C      NOMOD IS TOTAL NO. MODES
C
      WRITE(NOUT,60)
      DO 140 I=1,NTF
      MSNO=TMS(I,1)
      NCODE=TMS(I,2)
      DCS =TMS(I,3)
      MODE=TMS(I,4)
      WRITE(NOUT,40)I,MSNO,NCODE,DCS,MODE
      DO 110 J=1,NM
      JJ=J+NO
  110 PHIS(I,J) = PHI(MSNO,JJ,NCODE)
  140 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TFBLIN(NTI,
     1 NTF,MXBLK,MXOBLK,ANUM,ADEN,NZER,FACT,NPRN)
C
      COMMON /ZZZ/CASE(34),TITLE(14),NIN,NOUT,KROW,LINES,IPRNT,NER
C
      DIMENSION ANUM(NTF,MXBLK,1)
      DIMENSION ADEN(NTF,MXBLK,1)
      DIMENSION TN(3),TD(3),TL(3)
      DATA BLANK/      /
C
   10 FORMAT(3F12.0,3(3A4),T37,3F12.0)
   20 FORMAT(1H ,3(I4,2X),3(4X,3A4))
   25 FORMAT(1HO, TRANSFER FUNCTION BLOCK/ELEMENT POLYNOMIAL INPUT /1HO,
     1  TFNO  BLKNO ORDER  ,12X, NUM ,12X, DEN ,6X, NTFL )
C
      IF(FACT.EQ.0)FACT=1.0
      IF(NZER.NE.0)GO TO 90
      IFIN=NTF*MXBLK*MXOBLK
      DO 80 I=1,IFIN
      ANUM(I,1,1)=0.
   80 ADEN(I,1,1)=0.
C
   90 CONTINUE
C
C     NTFN = TF NO.
C     NORD = ORDER OF ELEMENT IN BLOCK
C     NBLK = BLOCK NO.
C     AN = NUM COEF
C     AD = DEN COEF
C
      IF(NPRN.NE.0)WRITE(NOUT,25)
  100 CONTINUE
      NTFL=0
      READ(NTI,10)ATFN,ABLK,AORD,TN,TD,TL,AN,AD,TLAST
      NTFN=ATFN
      NBLK=ABLK
      NORD=AORD
      IF(NTFN.LT.0)GO TO 144
      IF(NPRN.NE.0)
     1WRITE(NOUT,20)NTFN,NBLK,NORD,TN,TD,TL
      NORD=NORD+1
      IF(NTFN.GT.NTF)GO TO 730
      IF(NTFN.LT.1)GO TO 730
      IF(NBLK.GT.MXBLK)GO TO 725
      IF(NORD.GT.MXOBLK)GO TO 735
      IF(TL(1).NE.BLANK.OR.TL(2).NE.BLANK.OR.TL(3).NE.BLANK)NTFL=TLAST
      IF(NTFL.NE.0)GO TO 120
      IF(TN(1).NE.BLANK.OR.TN(2).NE.BLANK.OR.TN(3).NE.BLANK)
     1ANUM(NTFN,NBLK,NORD)=AN*FACT
      IF(TD(1).NE.BLANK.OR.TD(2).NE.BLANK.OR.TD(3).NE.BLANK)
     1ADEN(NTFN,NBLK,NORD)=AD*FACT
      GO TO 100
```

```
C                                                                      TFBLIN54
  120 CONTINUE                                                         TFBLIN55
C     PUT THIS TF BLOCK IN ALL DESIGNATED TF BLOCKS                    TFBLIN56
C                                                                      TFBLIN57
      IF(NTFL.GT.NTF)NTFL=NTF                                          TFBLIN58
      DO 130 N=NTFN,NTFL                                               TFBLIN59
      IF(TN(1).NE.BLANK.OR.TN(2).NE.BLANK.OR.TN(3).NE.BLANK)           TFBLIN60
     1ANUM(N,NBLK,NORD)=AN*FACT                                        TFBLIN61
      IF(TD(1).NE.BLANK.OR.TD(2).NE.BLANK.OR.TD(3).NE.BLANK)           TFBLIN62
     1ADEN(N,NBLK,NORD)=AD*FACT                                        TFBLIN63
  130 CONTINUE                                                         TFBLIN64
      GO TO 100                                                        TFBLIN65
C                                                                      TFBLIN66
  144 CONTINUE                                                         TFBLIN67
C                                                                      TFBLIN68
C                                                                      TFBLIN69
      RETURN                                                           TFBLIN70
C                                                                      TFBLIN71
  735 CONTINUE                                                         TFBLIN72
      WRITE(NOUT,736)                                                  TFBLIN73
  736 FORMAT(1H , ORDER OF INPUT ELEMENT EXCEEDS MAX - SOLN HALTS )    TFBLIN74
      STOP                                                             TFBLIN75
C                                                                      TFBLIN76
  730 CONTINUE                                                         TFBLIN77
      WRITE(NOUT,731)                                                  TFBLIN78
  731 FORMAT(1H , INPUT TF NO EXCEEDS MAX OR IS 0 - SOLN PROCEEDS )    TFBLIN79
      GO TO 100                                                        TFBLIN80
C                                                                      TFBLIN81
  725 CONTINUE                                                         TFBLIN82
      WRITE(NOUT,726)                                                  TFBLIN83
  726 FORMAT(1H , INPUT BLOCK NO EXCEEDS MAX - SOLN PROCEEDS )         TFBLIN84
      GO TO 100                                                        TFBLIN85
C                                                                      TFBLIN86
C                                                                      TFBLIN87
      END                                                              TFBLIN88
```

```
      SUBROUTINE TFBLOK(                                                TFBLOK 2
     1 NTF,MXBLK,MXOBLK,TFC,ANUM,ADEN,ATFN,ATFD,      ANUMT,ADENT,      TFBLOK 3
     1 MXORDN,MXORDD,NORDM)                                             TFBLOK 4
C                                                                       TFBLOK 5
      COMMON /ZZZ/CASE(34),TITLE(14),NIN,NOUT,KROW,LINES,IPRNT,NER      TFBLOK 6
C                                                                       TFBLOK 7
      DIMENSION TFC(2,NTF,1)                                            TFBLOK 8
      DIMENSION ANUM(NTF,MXBLK,1)                                       TFBLOK 9
      DIMENSION ADEN(NTF,MXBLK,1)                                       TFBLOK10
      DIMENSION ATFN(2,1)                                               TFBLOK11
      DIMENSION ATFD(2,1)                                               TFBLOK12
      DIMENSION ANUMT(MXBLK,1)                                          TFBLOK13
      DIMENSION ADENT(MXBLK,1)                                          TFBLOK14
C                                                                       TFBLOK15
 10   FORMAT(1H )                                                       TFBLOK16
 21   FORMAT(1H ,3(I4,2X),16X,E16.8)                                    TFBLOK17
 20   FORMAT(1H ,3(I4,2X),E16.8)                                        TFBLOK18
 30   FORMAT(1H0, TRANSFER FUNCTION POLYNOMIAL /1H0,                    TFBLOK19
     1  TFNO  ORDER  CODE    ,11X, REAL ,12X, IMAG )                    TFBLOK20
 40   FORMAT(1H0, SUMMARY ALL BLOCK INPUT TF DATA /1H0,                 TFBLOK21
     1 3( TFNBLKORD  ,4X, NUM ,8X, DEN ,4X))                            TFBLOK22
 45   FORMAT(1H ,3(3I3,1X,2E11.3))                                      TFBLOK23
C                                                                       TFBLOK24
C     CODE= 1,NUM COEF  , = 2,DEN COEF                                  TFBLOK25
C                                                                       TFBLOK26
      IF(NZER.NE.0)GO TO 90                                             TFBLOK27
      MXORDN=0                                                          TFBLOK28
      MXORDD=0                                                          TFBLOK29
C                                                                       TFBLOK30
 90   CONTINUE                                                          TFBLOK31
C                                                                       TFBLOK32
      IF(IPRNT.EQ.0)GO TO 120                                           TFBLOK33
      CALL HEADNG                                                       TFBLOK34
      WRITE(NOUT,40)                                                    TFBLOK35
      ICL=5                                                             TFBLOK36
      NF=1                                                              TFBLOK37
      NL=3                                                              TFBLOK38
 101  CONTINUE                                                          TFBLOK39
      IF(NL.GT.NTF)NL=NTF                                               TFBLOK40
      DO 110 J=1,MXBLK                                                  TFBLOK41
      DO 110 K=1,MXOBLK                                                 TFBLOK42
      KK=K-1                                                            TFBLOK43
      ICL=ICL+1                                                         TFBLOK44
      IF(ICL.LT.KROW)GO TO 105                                          TFBLOK45
      CALL HEADNG                                                       TFBLOK46
      WRITE(NOUT,40)                                                    TFBLOK47
      ICL=5                                                             TFBLOK48
 105  CONTINUE                                                          TFBLOK49
 110  WRITE(NOUT,45)(N,J,KK,ANUM(N,J,K),ADEN(N,J,K),N=NF,NL)            TFBLOK50
      WRITE(NOUT,10)                                                    TFBLOK51
      ICL=ICL+1                                                         TFBLOK52
      IF(NL.GE.NTF)GO TO 120                                            TFBLOK53
```

```
      NF=NL+1                                                           TFBLOK54
      NL=NL+3                                                           TFBLOK55
      GO TO 101                                                         TFBLOK56
  120 CONTINUE                                                          TFBLOK57
C                                                                       TFBLOK58
      CALL HEADNG                                                       TFBLOK59
      WRITE(NOUT,30)                                                    TFBLOK60
      ICL=5                                                             TFBLOK61
C                                                                       TFBLOK62
      NORDMT=2*NORDM                                                    TFBLOK63
      DO 1000 N=1,NTF                                                   TFBLOK64
C                                                                       TFBLOK65
C     TEST          DEN FOR ZEROS                                       TFBLOK66
      DO 600 J=1,MXBLK                                                  TFBLOK67
      ICHKD=0                                                           TFBLOK68
      DO 500 K=1,MXOBLK                                                 TFBLOK69
      ANUMT(J,K)=ANUM(N,J,K)                                            TFBLOK70
      ADENT(J,K)=ADEN(N,J,K)                                            TFBLOK71
      IF(ADENT(J,K).NE.0)ICHKD=1                                        TFBLOK72
  500 CONTINUE                                                          TFBLOK73
      IF(ICHKD.EQ.0)ADENT(J,1)=1.0                                      TFBLOK74
  600 CONTINUE                                                          TFBLOK75
C                                                                       TFBLOK76
      DO 145 J=1,NORDMT                                                 TFBLOK77
      ATFN(J,1)=0.                                                      TFBLOK78
  145 ATFD(J,1)=0.                                                      TFBLOK79
      ATFN(1,1)=1.0                                                     TFBLOK80
      ATFD(1,1)=1.0                                                     TFBLOK81
C                                                                       TFBLOK82
      DO 260 M=1,2                                                      TFBLOK83
      DO 260 K=1,NORDM                                                  TFBLOK84
  260 TFC(M,N,K)=0.                                                     TFBLOK85
C                                                                       TFBLOK86
C                                                                       TFBLOK87
      DO 160 J=1,MXBLK                                                  TFBLOK88
      DO 150 L=1,MXOBLK                                                 TFBLOK89
      KK=0                                                              TFBLOK90
      DO 150 K=L,NORDM                                                  TFBLOK91
      KK=KK+1                                                           TFBLOK92
      ATFN(2,K)=ATFN(2,K)+ANUMT( J,L)*ATFN(1,KK)                        TFBLOK93
      ATFD(2,K)=ATFD(2,K)+ADENT( J,L)*ATFD(1,KK)                        TFBLOK94
  150 CONTINUE                                                          TFBLOK95
      DO 155 K=1,NORDM                                                  TFBLOK96
      ATFN(1,K)=ATFN(2,K)                                               TFBLOK97
      ATFD(1,K)=ATFD(2,K)                                               TFBLOK98
      ATFN(2,K)=0.                                                      TFBLOK99
  155 ATFD(2,K)=0.                                                      TFBLO100
  160 CONTINUE                                                          TFBLO101
C                                                                       TFBLO102
C                                                                       TFBLO103
C                                                                       TFBLO104
      DO 300 K=1,NORDM,4                                                TFBLO105
```

```
      IF(ATFN(1,K).NE.0.AND.K.GT.MXORDN)MXORDN=K                        TFBL0106
      IF(ATFD(1,K).NE.0.AND.K.GT.MXORDD)MXORDD=K                        TFBL0107
      TFC(1,N,K)=ATFN(1,K)                                              TFBL0108
      TFC(2,N,K)=ATFD(1,K)                                              TFBL0109
  300 CONTINUE                                                          TFBL0110
      DO 310 K=2,NORDM,4                                                TFBL0111
      IF(ATFN(1,K).NE.0.AND.K.GT.MXORDN)MXORDN=K                        TFBL0112
      IF(ATFD(1,K).NE.0.AND.K.GT.MXORDD)MXORDD=K                        TFBL0113
      TFC(1,N,K)=ATFN(1,K)                                              TFBL0114
      TFC(2,N,K)=ATFD(1,K)                                              TFBL0115
  310 CONTINUE                                                          TFBL0116
      IF(NORDM.LT.3)GO TO 350                                           TFBL0117
      DO 320 K=3,NORDM,4                                                TFBL0118
      IF(ATFN(1,K).NE.0.AND.K.GT.MXORDN)MXORDN=K                        TFBL0119
      IF(ATFD(1,K).NE.0.AND.K.GT.MXORDD)MXORDD=K                        TFBL0120
      TFC(1,N,K)=-ATFN(1,K)                                             TFBL0121
      TFC(2,N,K)=-ATFD(1,K)                                             TFBL0122
  320 CONTINUE                                                          TFBL0123
      IF(NORDM.LT.4)GO TO 350                                           TFBL0124
      DO 330 K=4,NORDM,4                                                TFBL0125
      IF(ATFN(1,K).NE.0.AND.K.GT.MXORDN)MXORDN=K                        TFBL0126
      IF(ATFD(1,K).NE.0.AND.K.GT.MXORDD)MXORDD=K                        TFBL0127
      TFC(1,N,K)=-ATFN(1,K)                                             TFBL0128
      TFC(2,N,K)=-ATFD(1,K)                                             TFBL0129
  330 CONTINUE                                                          TFBL0130
C                                                                       TFBL0131
  350 CONTINUE                                                          TFBL0132
C                                                                       TFBL0133
      IF(IPRNT.EQ.0)GO TO 1000                                          TFBL0134
C                                                                       TFBL0135
      L=1                                                               TFBL0136
      DO 400 K=1,MXORDN,2                                               TFBL0137
      KK=K-1                                                            TFBL0138
      ICL=ICL+1                                                         TFBL0139
      IF(ICL.LT.KROW)GO TO 395                                          TFBL0140
      CALL HEADNG                                                       TFBL0141
      WRITE(NOUT,30)                                                    TFBL0142
      ICL=5                                                             TFBL0143
  395 CONTINUE                                                          TFBL0144
      WRITE(NOUT,20)N,KK,L,TFC(L,N,K)                                   TFBL0145
      WRITE(NOUT,21)N,K,L,TFC(L,N,K+1)                                  TFBL0146
  400 CONTINUE                                                          TFBL0147
C                                                                       TFBL0148
      L=2                                                               TFBL0149
      DO 410 K=1,MXORDD,2                                               TFBL0150
      KK=K-1                                                            TFBL0151
      ICL=ICL+1                                                         TFBL0152
      IF(ICL.LT.KROW)GO TO 405                                          TFBL0153
      CALL HEADNG                                                       TFBL0154
      WRITE(NOUT,30)                                                    TFBL0155
      ICL=5                                                             TFBL0156
  405 CONTINUE                                                          TFBL0157
```

```
      WRITE(NOUT,20)N,KK,L,TFC(L,N,K)                                   TFBL0158
      WRITE(NOUT,21)N,K,L,TFC(L,N,K+1)                                  TFBL0159
  410 CONTINUE                                                          TFBL0160
C                                                                       TFBL0161
      WRITE(NOUT,10)                                                    TFBL0162
      ICL=ICL+1                                                         TFBL0163
C                                                                       TFBL0164
 1000 CONTINUE                                                          TFBL0165
C                                                                       TFBL0166
C                                                                       TFBL0167
      RETURN                                                            TFBL0168
      END                                                               TFBL0169
```

```
      SUBROUTINE TRFVAL(NTF,A   ,OMEGA,MXORDN,MXORDD)                    TRFVAL 2
C                                                                        TRFVAL 3
      COMMON /ZZZ/CASE(34),TITLE(14),NIN,NOUT,KROW,LINES,IPRNT,NER       TRFVAL 4
C                                                                        TRFVAL 5
      DIMENSION A(2,NTF,1)                                               TRFVAL 6
C                                                                        TRFVAL 7
   50 FORMAT(1HO, TRANSFER FUNCTIONS FOR 1 RAD/SEC /1HO,                 TRFVAL 8
     1  TFNC ,6X, RE(NUM) ,6X, IM(NUM) ,6X, RE(DEN) ,6X, IM(DEN) ,       TRFVAL 9
     2  7X, RE(TF) , 7X, IM(TF) ,10X, MOD ,8X, PHASE )                   TRFVAL10
   60 FORMAT(1H ,I4          ,2X,7E13.5,5X,F8.3)                         TRFVAL11
C                                                                        TRFVAL12
      WRITE(NOUT,50)                                                     TRFVAL13
      DO 220 N=1,NTF                                                     TRFVAL14
      AR=0.                                                              TRFVAL15
      AI=0.                                                              TRFVAL16
      BR=1.                                                              TRFVAL17
      BI=0.                                                              TRFVAL18
      IF(MXORDN.EQ.0)GO TO 600                                           TRFVAL19
      DO 300 I=1,MXORDN,2                                                TRFVAL20
      NORD=I-1                                                           TRFVAL21
      POWR=OMEGA**NORD                                                   TRFVAL22
      POWI=OMEGA**I                                                      TRFVAL23
      AR=AR+A(1,N,I)*POWR                                                TRFVAL24
      AI=AI+A(1,N,I+1)*POWI                                              TRFVAL25
  300 CONTINUE                                                           TRFVAL26
C                                                                        TRFVAL27
      IF(MXORDD.EQ.0)GO TO 600                                           TRFVAL28
      BR=0.                                                              TRFVAL29
      DO 500 I=1,MXORDD,2                                                TRFVAL30
      NORD=I-1                                                           TRFVAL31
      POWR=OMEGA**NORD                                                   TRFVAL32
      POWI=OMEGA**I                                                      TRFVAL33
      BR=BR+A(2,N,I)*POWR                                                TRFVAL34
      BI=BI+A(2,N,I+1)*POWI                                              TRFVAL35
  500 CONTINUE                                                           TRFVAL36
C                                                                        TRFVAL37
  600 CONTINUE                                                           TRFVAL38
      DEN=BR**2+BI**2                                                    TRFVAL39
      TFR=(AR*BR+AI*BI)/DEN                                              TRFVAL40
      TFI=(AI*BR-AR*BI)/DEN                                              TRFVAL41
C                                                                        TRFVAL42
      DEN=SQRT(TFR**2+TFI**2)                                            TRFVAL43
      TAN=0.0                                                            TRFVAL44
      IF(TFR.NE.0)GO TO 200                                              TRFVAL45
      IF(TFI.LT.0)TAN=-1.57080                                           TRFVAL46
      IF(TFI.GT.0)TAN= 1.57080                                           TRFVAL47
      GO TO 210                                                          TRFVAL48
  200 CONTINUE                                                           TRFVAL49
      TAN=ATAN2(TFI,TFR)                                                 TRFVAL50
  210 CONTINUE                                                           TRFVAL51
      TAN=TAN*57.29578                                                   TRFVAL52
      WRITE(NOUT,60)N,AR ,AI ,BR ,BI ,TFR,TFI,DEN,TAN                    TRFVAL53
```

```
  220 CONTINUE                                                  TRFVAL54
C                                                               TRFVAL55
      RETURN                                                    TRFVAL56
      END                                                       TRFVAL57
```

```
      SUBROUTINE CFREQR (CR,AMODNO,FREQIN,EMBAR,CAY,CPXDPG,NM,NVBW,NG,
     2   TMSS,TMSA,TFCS,TFCA,PHISS,PHISA
     3  )
C
      DIMENSION CR(3,8)
C
      DIMENSION AMODNO(20)
      DIMENSION FREQIN(NM),EMBAR(NM,NM),CAY(NM),CPXDPG(NM)
      DIMENSION TMSS(1),TMSA(1),TFCS(1),TFCA(1),PHISS(1),PHISA(1)
      COMMON NAA,A(1)
      COMMON/XTF/XF(100)
      EQUIVALENCE (XF(70),MXORD)
     5 , (XF(71),MXORSN),(XF(72),MXORSD),(XF(73),MXORAN)
     6 , (XF(74),MXORAD),(XF(75),NTFS),(XF(76),NTFA)
     1 , (XF(6),NSYM),(XF(7),NASYM)
     8 , (XF(43),NBOX), (XF(44),NSBETO)
C
      NMMX = MAXO(NSYM,NASYM)
C
      NAAO = NAA
      L12 = NAA
      L13 = L12 + 2*NMMX*NMMX
      L14 = L13 + 2*NMMX
      L15 = L14 + NVBW
      L18 = L15 + 2*NM
      L19 = L18 + 2*NMMX*NMMX
      L20 = L19 + (NBOX+2*NSBETO)*NSYM
      NAA = L20 + (NBOX+2*NSBETO)*NASYM
      CALL FREQRS (CR,AMODNO,FREQIN,EMBAR,CAY,CPXDPG,NM
     1 , A(L12),A(L13),A(L14),A(L15),NVBW
     2 , A(L18),
     3 TFCS,TFCA,TMSS,TMSA,PHISS,PHISA,NTFS,NTFA,
     4   MXORSN,MXORAN,MXORSD,MXORAD
     5 , A(L19),A(L20),NBOX,NSBETO)
      NAA = NAAO
      RETURN
      END
```

```
      SUBROUTINE COEFF(RKIN,RK,NK,C,COEF)
      DIMENSION RK(1),C(13,10),COEF(1)
      REAL KF(13)
C
C *** FORM KF
C
      NK3 = NK+3
      KF(1) = 1.0
      KF(2) = RKIN
      KF(3) = 0.0
      DO 10 J=4,NK3
      A = ( RKIN-RK(J-3) )**2
   10 KF(J) = A*ALOG(A)
C
      DO 20 J=1,NK
      COEF(J) = 0.0
      DO 20 I=1,NK3
   20 COEF(J) = COEF(J) + KF(I) * C(I,J)
C
      RETURN
      END
```

```
      SUBROUTINE FORMC (RK,NK,W,IPRN)                                   FORMC   2
      DIMENSION RK(1),W(13,10)                                          FORMC   3
      REAL KD(13,13)                                                    FORMC   4
      NK3 = NK+3                                                        FORMC   5
      DO 30 I=1,NK3                                                     FORMC   6
      DO 10 J=1,NK3                                                     FORMC   7
   10 KD(I,J) = 0.0                                                     FORMC   8
      DO 20 J=1,NK                                                      FORMC   9
   20 W(I,J) = 0.0                                                      FORMC  10
   30 CONTINUE                                                          FORMC  11
      DO 40 J=1,NK                                                      FORMC  12
      I = J+3                                                           FORMC  13
   40 W(I,J) = 1.0                                                      FORMC  14
      KD(3,3) = 1.0                                                     FORMC  15
      DO 60 J=4,NK3                                                     FORMC  16
      KD(1,J) = 1.0                                                     FORMC  17
      KD(J,1) = 1.0                                                     FORMC  18
      KD(2,J) = RK(J-3)                                                 FORMC  19
      KD(J,2) = RK(J-3)                                                 FORMC  20
      DO 50 I=4,NK3                                                     FORMC  21
      IF (I.EQ.J) GO TO 50                                              FORMC  22
      A = ( RK(I-3)-RK(J-3) )**2                                        FORMC  23
      KD(I,J) = A*ALOG(A)                                               FORMC  24
      KD(J,I) = KD(I,J)                                                 FORMC  25
   50 CONTINUE                                                          FORMC  26
   60 KD(J,J) = 0.0                                                     FORMC  27
      D = 0.0                                                           FORMC  28
      CALL MIS1(KD,NK3,13,W,NK,NERR,D)                                  FORMC  29
      IF(IPRN.GT.1)                                                     FORMC  30
     1  CALL PRNT(W,NK3,NK,13,1,8HW MATRIX,2)                           FORMC  31
      RETURN                                                            FORMC  32
      END                                                               FORMC  33
```

```
      SUBROUTINE FREQRS(                                                FREQRS 2
     1   CP,AMCDNO,FREQIN,EMBAR,CAY,CPXDPG,NMD                          FREQRS 3
     2 , D,F,OMEGA,Q,                                                   FREQRS 4
     5 NVBW,TFM,TFCS,TFCA,TMSS,TMSA,PHISS,PHISA,NTFS,NTFA,              FREQRS 5
     6 MXORSN,MXORAN,MXORSD,MXORAD,                                     FREQRS 6
     7  SPLS,SPLA,NBCX,NSBETO)                                          FREQRS 7
C                                                                       FREQRS 8
      COMMON NAA,A(1)                                                   FREQRS 9
C                                                                       FREQRS10
      COMMON/ZZZ/CASE(48),NIN,NOUT,KROW,LINES, IPRNT,NER                FREQRS11
      COMMON/XTF/NFLAGS(100)                                            FREQRS12
C                                                                       FREQRS13
      EQUIVALENCE  (NFLAGS(6),NSYM),(NFLAGS(7),NASYM)                   FREQRS14
      EQUIVALENCE  (NFLAGS(11),IDENT)                                   FREQRS15
      EQUIVALENCE  (NFLAGS(12),BZERO),(NFLAGS(13),RHOO)                 FREQRS16
     1 , (NFLAGS(14),SIGMA ),(NFLAGS(15),EQUVAS),(NFLAGS(16),IUNITS)    FREQRS17
     2 , (NFLAGS(17),VKEAS),  (NFLAGS(18),VTFPS)                        FREQRS18
     3 , (NFLAGS(20),KPRCHK),(NFLAGS(21),IPREQ  )                       FREQRS19
     4 , (NFLAGS(19),ALT)                                               FREQRS20
     5 , (NFLAGS(100),SIZFCT)                                           FREQRS21
     6        ,        (NFLAGS(42),NG)                                  FREQRS22
     7 , (NFLAGS(1),AMACH)                                              FREQRS23
C                                                                       FREQRS24
      DIMENSION IHD(50),  RHD(50)                                       FREQRS25
      EQUIVALENCE  (IHD(1),RHD(1))                                      FREQRS26
      DIMENSION UNITS(3)                                                FREQRS27
      DIMENSION VELUN(3)                                                FREQRS28
      DIMENSION CR(3,8)                                                 FREQRS29
      DIMENSION AMCDNO(20)                                              FREQRS30
      DIMENSION FREQIN(1)                                               FREQRS31
      DIMENSION EMBAR(NMD,NMD)                                          FREQRS32
      DIMENSION CAY(1)                                                  FREQRS33
      DIMENSION CPXDPG(1)                                               FREQRS34
      DIMENSION D(2,1)                                                  FREQRS35
      DIMENSION F(2,1)                                                  FREQRS36
      DIMENSION SPLS(1),SPLA(1)                                         FREQRS37
      DIMENSION Q(2,NMD)                                                FREQRS38
      DIMENSION OMEGA(1)                                                FREQRS39
      DIMENSION TFM(1)                                                  FREQRS40
      DIMENSION TFCS(1),TFCA(1),PHISS(1),PHISA(1)                       FREQRS41
      DIMENSION TMSS(NTFS,1),TMSA(NTFA,1)                               FREQRS42
      DIMENSION RK(10),C(13,10)                                         FREQRS43
      DIMENSION COEF(10),VOBWI(10), IVBWI(10)                           FREQRS44
C                                                                       FREQRS45
      COMMON/DISK2/ND2,ITBL2(843),NRECSA,IBUMP,NKD,VOBWS(20)            DISK2  2
      COMMON/AEROMX/INTARO,NVBWMX,VOBWIN(400),RINTP(50,3)               FREQRS47
C                                                                       FREQRS48
   20 FORMAT(1HO,20X, GUST ORIENTATION NO. ,I4/1H ,                     FREQRS49
     1 20X, GAMX =    ,F8.5/1H ,                                        FREQRS50
     1 20X, GAMY =    ,F8.5/1H ,                                        FREQRS51
     1 20X, GAMZ =    ,F8.5)                                            FREQRS52
   30 FORMAT(1H ,I4,2E16.8)                                             FREQRS53
```

```
10  FORMAT(1H0,20X, ACS ON FOR THIS ANALYSIS )                              FREQRS54
40  FORMAT(1H0,*IVBW = *,I3,5X    ,*VOBW = *,F10.5,5X,                      FREQRS55
   1  FREQ =   ,F10.4)                                                       FREQRS56
45  FORMAT(1H , IVBW =   ,I3,5X, VOBW =   ,F10.5,5X, FREQ =   ,F10.4)       FREQRS57
50  FORMAT(1H ,20X,*FREQUENCY RESPONSE SOLUTIONS*/1H0,*VEL  =*,F8.2,         FREQRS58
   1 1X,A4,5X, DYNP =  ,F10.2,5X, Q/V =  ,E12.5,                             FREQRS59
   2 10X, ALTITUDE =   ,F9.2,   FT. /1H0,10X, Q    GENERALIZED RESPONSE )FREQRS60
60  FORMAT(1H0,30X,*IVBW = *,I3,5X,*FREQ = *,F10.4)                         FREQRS61
100 FORMAT(1H1,20X,*HEY  YOU CAN NOT  INTERP WITH LESS THAN 3 AICS*)        FREQRS62
175 FORMAT(33H0********************************,                            FREQRS63
   1 * WARNING - EXTRAPOLATION OCCURRED GETTING THIS AERO SET *,            FREQRS64
   1 20H********************)                                                FREQRS65
C                                                                            FREQRS66
    DATA UNITS/1.0,0.59208578,0.68181818/                                   FREQRS67
    DATA VELUN/4H FPS,4HKEAS,4HMMPH/                                        FREQRS68
    DATA TWOPI/6.2831853/                                                   FREQRS69
    DATA COREFQ/8H FREQRS /                                                 FREQRS70
C                                                                            FREQRS71
C                                                                            FREQRS72
C   IPREQ=1 PRINTS Q   KPRCHK=1 PRINTS MATS   KPRCHK=2 PRINTS MORE MATFREQRS73
C                                                                            FREQRS74
    ITEMPQ=IPREQ                                                            FREQRS75
    ITEMPK=KPRCHK                                                           FREQRS76
    IPRQ1=ITEMPQ                                                            FREQRS77
    IPRQ2=ITEMPQ                                                            FREQRS78
    IPRK1=ITEMPK                                                            FREQRS79
    IPRK2=ITEMPK                                                            FREQRS80
C                                                                            FREQRS81
C                                                                            FREQRS82
C   ISETA=1 FOR INC. K ON AERO TAPE, =-1 FOR DEC. K                         FREQRS83
C                                                                            FREQRS84
    ISETA=1                                                                 FREQRS85
    IF (VOBWS(NKD).GT.VOBWS(1)) ISETA=-1                                    FREQRS86
    VOBWUP=VOBWS(NKD)                                                       FREQRS87
    VOBWLO=VOBWS(1)                                                         FREQRS88
    IF (NKD.LT.3) GO TO 1090                                                FREQRS89
    IF(ISETA.GT.0)VOBWUP=VOBWS(1)                                           FREQRS90
    IF (ISETA.GT.0) VOBWLO=VOBWS(NKD)                                       FREQRS91
    IUNIT=IABS(IUNITS)                                                      FREQRS92
    FACTOR=SIZFCT**2                                                        FREQRS93
    OMEGF=VTFPS/(BZERO*SIZFCT/12.)                                          FREQRS94
    RHO = RHOO*SIGMA                                                        FREQRS95
    DYNP=RHO*VTFPS**2*FACTOR/288.                                           FREQRS96
    VWG=DYNP/VTFPS                                                          FREQRS97
    AMULTA=-DYNP                                                            FREQRS98
C   ALL ANAL IS FOR A 1 FPS GUST                                            FREQRS99
C                                                                            FREQR100
C                                                                            FREQR101
C   MAX INPUT FREQ S/B ABOUT 5*MAX ELASTIC MODE FREQ,EXCL OF CONTROL SFREQR102
C                                                                            FREQR103
C                                                                            FREQR104
    NK = MINO(10,NKD)                                                       FREQR105
```

```
C                                                                       FREQR106
      NTOTAP = NBOX + 2*NSBETO                                          FREQR107
C                                                                       FREQR108
C     READ BACK GEN GUST INTEGRATION MATS                               FREQR109
C                                                                       FREQR110
      NW=NTOTAP*NSYM                                                    FREQR111
      LOC=NKD*IBUMP+1                                                   FREQR112
      CALL READMS(ND2,SPLS,NW,LOC)                                      FREQR113
      NW=NTOTAP*NASYM                                                   FREQR114
      LOC=LOC+1                                                         FREQR115
      CALL READMS(ND2,SPLA,NW,LOC)                                      FREQR116
C                                                                       FREQR117
C                                                                       FREQR118
      DO 240 IVBW=1,NVBW                                                FREQR119
      FREQ=VOBWIN(IVBW)                                                 FREQR120
      OMEGA(IVBW)=TWOPI*FREQ                                            FREQR121
      IF(OMEGA(IVBW).LE.0)OMEGA(IVBW)=1.E-06                            FREQR122
  240 CONTINUE                                                          FREQR123
C                                                                       FREQR124
      DO 190 I=1,50                                                     FREQR125
  190 IHD(I) = 0                                                        FREQR126
      IHD(2) = IDENT                                                    FREQR127
      IHD(3) = NG                                                       FREQR128
      IHD(4) = NVBW                                                     FREQR129
      IHD(5) = NSYM                                                     FREQR130
      IHD(6) = NASYM                                                    FREQR131
      IHD(7) = NK                                                       FREQR132
      IHD(8)=NTOTAP                                                     FREQR133
      RHD(11) = VTFPS                                                   FREQR134
      RHD(12) = SIGMA                                                   FREQR135
      CALL WRFRSP (1,0,0,IHD,NOUT,NER)                                  FREQR136
      CALL WRFRSP (2,0,0,CR,NOUT,NER)                                   FREQR137
      CALL WRFRSP (3,0,0,OMEGA,NOUT,NER)                                FREQR138
C                                                                       FREQR139
C                                                                       FREQR140
      L1S = 2*NSYM*NSYM*NK                                              FREQR141
      L1A = 2*NASYM*NASYM*NK                                            FREQR142
      L2S = 2*NTOTAP*NK                                                 FREQR143
      L2A = 2*NTOTAP*NK                                                 FREQR144
      L3S = NTOTAP*NK                                                   FREQR145
      L3A = NTOTAP*NK                                                   FREQR146
      L4S = 2*NTOTAP                                                    FREQR147
      L4A = 2*NTOTAP                                                    FREQR148
      N1S = NAA                                                         FREQR149
      N2S = N1S + L1S                                                   FREQR150
      N3S = N2S + L2S                                                   FREQR151
      N4S = N3S + L3S                                                   FREQR152
      N5S = N4S + L3S                                                   FREQR153
      N1A = N5S + L4S                                                   FREQR154
      N2A = N1A + L1A                                                   FREQR155
      N3A = N2A + L2A                                                   FREQR156
      N4A = N3A + L3A                                                   FREQR157
```

```
      N5A = N4A + L3A                                                   FREQR158
      NTLG = N5A + L4A                                                  FREQR159
      WRITE(NOUT,999)NTLG,CORERQ                                        FREQR160
  999 FORMAT(1H0,20X,I10,  WORDS OF CORE RQD FOR STEP +++ ,A10, +++ )   FREQR161
      IMOS = 0                                                          FREQR162
      IMOA = 0                                                          FREQR163
C                                                                       FREQR164
C                                                                       FREQR165
C     START GUST LOOP                                                   FREQR166
C                                                                       FREQR167
      DO 900 IG=1,NG                                                    FREQR168
C                                                                       FREQR169
      IRECGS=0                                                          FREQR170
      IRECGA=0                                                          FREQR171
      IZROS=0                                                           FREQR172
      IZROA=0                                                           FREQR173
C     IZROS=1 INDICATES NO SYMMETRIC GUST FORCES THIS ORIENTATION       FREQR174
C     IZROA=1 INDICATES NO ANTISYMMETRIC GUST FORCES THIS ORIENTATION   FREQR175
C     IRECGS = 0 SAYS READ SYM GUST COLS AND SET UP INTERP              FREQR176
C     IRECGA = 0 SAYS READ ASM GUST COLS AND SET UP INTERP              FREQR177
C     IMOS = 0 SAYS DO SAME FOR SYM MOTION DEPENDENT AERO               FREQR178
C     IMOA = 0 SAYS DO SAME FOR ASM MOTION DEPENDENT AERO               FREQR179
C                                                                       FREQR180
      CALL HEADNG                                                       FREQR181
      WRITE(NOUT,20)IG,(CR(L,IG),L=1,3)                                 FREQR182
      WRITE(NOUT,50)VKEAS,VELUN(IUNIT),DYNP,VWG,ALT                     FREQR183
      LINES=LINES+7                                                     FREQR184
C                                                                       FREQR185
C     START FREQUENCY LOOP                                              FREQR186
C                                                                       FREQR187
      DO 800 IVBW=1,NVBW                                                FREQR188
C                                                                       FREQR189
      NM = NSYM                                                         FREQR190
      M = 0                                                             FREQR191
      IND = 1                                                           FREQR192
      N1 = N1S                                                          FREQR193
      N2 = N2S                                                          FREQR194
      N3 = N3S                                                          FREQR195
      N4 = N4S                                                          FREQR196
      N5 = N5S                                                          FREQR197
      IMO = IMOS                                                        FREQR198
      IRECG=IRECGS                                                      FREQR199
C                                                                       FREQR200
  300 CONTINUE                                                          FREQR201
C                                                                       FREQR202
      FREQ=VOBWIN(IVBW)                                                 FREQR203
      VOBW=OMEGF/OMEGA(IVBW)                                            FREQR204
C                                                                       FREQR205
C     IF THIS VOBW IS OUT OF RANGE OF INTERP, DONT USE IT               FREQR206
C                                                                       FREQR207
      IF(VOBW.GE.VOBWLO.AND.VOBW.LE.VOBWUP)GO TO 335                    FREQR208
C                                                                       FREQR209
```

```
      WRITE(NOUT,175)                                                   FREQR210
      LINES=LINES+2                                                     FREQR211
      DO 320 I=1,NM                                                     FREQR212
      F(1,I)=0.                                                         FREQR213
  320 F(2,I)=0.                                                         FREQR214
      I1=N5                                                             FREQR215
      I2=N5+2*NTOTAP-1                                                  FREQR216
      DO 321 I=I1,I2                                                    FREQR217
  321 A(I)=0.0                                                          FREQR218
      GO TO 590                                                         FREQR219
C                                                                       FREQR220
  335 CONTINUE                                                          FREQR221
C                                                                       FREQR222
      IPREQ=0                                                           FREQR223
      KPRCHK=0                                                          FREQR224
      IF(ITEMPK.NE.0.AND.IG.EQ.1.AND.IVBW.EQ.1)KPRCHK=1                 FREQR225
      IF(IG.EQ.IPRK1.OR.IG.EQ.IPRK2)KPRCHK=1                            FREQR226
      IF(IG.EQ.IPRQ1.OR.IG.EQ.IPRQ2)IPREQ=1                             FREQR227
C                                                                       FREQR228
      IF(IND.EQ.3.AND.IZROA.EQ.0)GO TO 290                              FREQR229
      IF(IND.EQ.1.AND.IZROS.EQ.0)GO TO 290                              FREQR230
C                                                                       FREQR231
      DO 210 I=1,NM                                                     FREQR232
      F(1,I)=0.                                                         FREQR233
  210 F(2,I)=0.                                                         FREQR234
      I1=N5                                                             FREQR235
      I2=N5+2*NTOTAP-1                                                  FREQR236
      DO 211 I=I1,I2                                                    FREQR237
  211 A(I)=0.0                                                          FREQR238
      GO TO 590                                                         FREQR239
C                                                                       FREQR240
  290 CONTINUE                                                          FREQR241
C                                                                       FREQR242
C                                                                       FREQR243
      CALL TDTINT(                                                      FREQR244
     1 VOBW,ISETA,IMC,D,A(N1),A(N2),NTOTAP,IG,COEF,VM,NK,KPRCHK,IND,    FREQR245
     3 VOBWI,IVBWI,IRECG,RK,C)                                          FREQR246
C                                                                       FREQR247
      IPREQ=0                                                           FREQR248
      KPRCHK=0                                                          FREQR249
      IF(IG.EQ.IPRQ1.OR.IG.EQ.IPRQ2)IPREQ=1                             FREQR250
      IF(IG.EQ.IPRK1.OR.IG.EQ.IPRK2)KPRCHK=1                            FREQR251
C                                                                       FREQR252
C                                                                       FREQR253
      ICACS=0                                                           FREQR254
      IF(IND.EQ.3)GO TO 331                                             FREQR255
      IF(NTFS.EQ.0)GO TO 332                                            FREQR256
      ICACS=1                                                           FREQR257
      CALL TFMATF(NTFS,NM,TFM,D,OMEGA(IVBW),TFCS,TMSS,0,                FREQR258
     1 MXORSN,MXORSD,PHISS,KPRCHK)                                      FREQR259
      GO TO 332                                                         FREQR260
  331 IF(NTFA.EQ.0)GO TO 332                                            FREQR261
```

```
      ICACS=1                                                           FREQR262
      CALL  TFMATF(NTFA,NM,TFM,D,OMEGA(IVBW),TFCA,TMSA,NSYM,            FREQR263
     1 MXORAN,MXORAD,PHISA,KPRCHK)                                      FREQR264
C                                                                       FREQR265
  332 CONTINUE                                                          FREQR266
C                                                                       FREQR267
      IF(IND.EQ.3)GO TO 333                                             FREQR268
C                                                                       FREQR269
      CALL GSTINI(IZROS,A(N5),                                          FREQR270
     1 SPLS,NSYM,COEF,                                                  FREQR271
     2 F,A(N2),NTOTAP,NK,A(N3),A(N4),IRECG,KPRCHK,IG)                   FREQR272
C                                                                       FREQR273
      GO TO 340                                                         FREQR274
C                                                                       FREQR275
  333 CONTINUE                                                          FREQR276
C                                                                       FREQR277
C                                                                       FREQR278
      CALL GSTINI(IZROA,A(N5),                                          FREQR279
     1 SPLA,NASYM,COEF,                                                 FREQR280
     2 F,A(N2),NTOTAP,NK,A(N3),A(N4),IRECG,KPRCHK,IG)                   FREQR281
C                                                                       FREQR282
  340 CONTINUE                                                          FREQR283
C                                                                       FREQR284
      IF(IND.EQ.3.AND.IZROA.EQ.0)GO TO 350                              FREQR285
      IF(IND.EQ.1.AND.IZROS.EQ.0)GO TO 350                              FREQR286
      DO 345 I=1,NM                                                     FREQR287
      F(1,I)=0.                                                         FREQR288
  345 F(2,I)=0.                                                         FREQR289
      I1=N5                                                             FREQR290
      I2=N5+2*NTOTAP-1                                                  FREQR291
      DO 346 I=I1,I2                                                    FREQR292
  346 A(I)=0.0                                                          FREQR293
      GO TO 590                                                         FREQR294
C                                                                       FREQR295
  350 CONTINUE                                                          FREQR296
C                                                                       FREQR297
       AMULT=-OMEGA(IVBW)**2                                            FREQR298
C                                                                       FREQR299
C     CALC HARMONIC GUST EQNS Z    F                                    FREQR300
      DO 405 I=1,NM                                                     FREQR301
      IM=I+M                                                            FREQR302
      F(1,I)=F(1,I)*VWG                                                 FREQR303
      F(2,I)=F(2,I)*VWG                                                 FREQR304
      DO 400 J=1,NM                                                     FREQR305
      JM=J+M                                                            FREQR306
      LOC=I+(J-1)*NM                                                    FREQR307
      D(1,LOC)=AMULTA*D(1,LOC)+AMULT*EMBAR(IM,JM)                       FREQR308
      D(2,LOC)=AMULTA*D(2,LOC)                                          FREQR309
  400 CONTINUE                                                          FREQR310
      LOC=I+(I-1)*NM                                                    FREQR311
      D(1,LOC)=D(1,LOC)+CAY(IM)                                         FREQR312
      D(2,LOC)=D(2,LOC)+CPXOPG(IM)*CAY(IM)                              FREQR313
```

```
  405 CONTINUE                                                        FREQR314
C                                                                     FREQR315
      IF(IND.EQ.3)GO TO 550                                           FREQR316
      IF(NTFS.EQ.0)GO TO 560                                          FREQR317
      CALL TFMATM(NTFS,NM,TFM,D,OMEGA(IVBW),TFCS,TMSS,0,              FREQR318
     1 MXORSN,MXORSD,PHISS,EMBAR,NMD,AMULT,KPRCHK)                    FREQR319
      GO TO 560                                                       FREQR320
  550 IF(NTFA.EQ.0)GO TO 560                                          FREQR321
      CALL TFMATM(NTFA,NM,TFM,D,OMEGA(IVBW),TFCA,TMSA,NSYM,           FREQR322
     1 MXORAN,MXORAD,PHISA,EMBAR,NMD,AMULT,KPRCHK)                    FREQR323
  560 CONTINUE                                                        FREQR324
C                                                                     FREQR325
C                                                                     FREQR326
C     NOW STRIKE OUT THE JIG MODES                                    FREQR327
C     (AND TRIM MODES UNLESS THEY ARE INCLUDED IN THE ELASTIC SETS)   FREQR328
C                                                                     FREQR329
      M1=7                                                            FREQR330
      M2=7                                                            FREQR331
      MSTP=1                                                          FREQR332
      M3=AMODNO(4)                                                    FREQR333
      M4=AMODNO(5)                                                    FREQR334
      M5=AMODNO(6)                                                    FREQR335
      IF(M5.LT.M3.OR.M5.GT.M4)M1=6                                    FREQR336
      IF(IND.EQ.1)GO TO 410                                           FREQR337
      M1=18                                                           FREQR338
      M2=18                                                           FREQR339
      MSTP=1                                                          FREQR340
      M3=AMODNO(14)                                                   FREQR341
      M4=AMODNO(15)                                                   FREQR342
      M5=AMODNO(17)                                                   FREQR343
      IF(M5.LT.M3.OR.M5.GT.M4)M1=17                                   FREQR344
      M5=AMODNO(16)                                                   FREQR345
      IF((M5.LT.M3.OR.M5.GT.M4).AND.M1.EQ.18)MSTP=2                   FREQR346
      IF(M5.LT.M3.OR.M5.GT.M4)M1=16                                   FREQR347
  410 DO 440 K=M1,M2,MSTP                                             FREQR348
      I=AMODNO(K)-M                                                   FREQR349
      IF(I.LE.0)GO TO 440                                             FREQR350
      F(1,I)=0.0                                                      FREQR351
      F(2,I)=0.0                                                      FREQR352
      DO 430 J=1,NM                                                   FREQR353
      LOC=I+(J-1)*NM                                                  FREQR354
      D(1,LOC)=0.0                                                    FREQR355
      D(2,LOC)=0.0                                                    FREQR356
      LOC=J+(I-1)*NM                                                  FREQR357
      D(1,LOC)=0.0                                                    FREQR358
      D(2,LOC)=0.0                                                    FREQR359
  430 CONTINUE                                                        FREQR360
      LOC=I+(I-1)*NM                                                  FREQR361
      D(1,LOC)=1.0                                                    FREQR362
  440 CONTINUE                                                        FREQR363
C                                                                     FREQR364
C     NOW ELIM ANY DELETED MODE FROM SOLN                             FREQR365
```

```
C                                                                       FREQR366
      M1=8                                                              FREQR367
      M2=9                                                              FREQR368
      IF(IND.EQ.1)GO TO 415                                             FREQR369
      M1=19                                                             FREQR370
      M2=20                                                             FREQR371
  415 DO 425 K=M1,M2                                                    FREQR372
      I=AMODNO(K)-M                                                     FREQR373
      IF(I.LE.0)GO TO 425                                               FREQR374
      F(1,I)=0.                                                         FREQR375
      F(2,I)=0.                                                         FREQR376
      DO 420 J=1,NM                                                     FREQR377
      LOC=I+(J-1)*NM                                                    FREQR378
      D(1,LOC)=0.                                                       FREQR379
      D(2,LOC)=0.                                                       FREQR380
      LOC=J+(I-1)*NM                                                    FREQR381
      D(1,LOC)=0.                                                       FREQR382
      D(2,LOC)=0.                                                       FREQR383
  420 CONTINUE                                                          FREQR384
      LOC=I+(I-1)*NM                                                    FREQR385
      D(1,LOC)=1.0                                                      FREQR386
  425 CONTINUE                                                          FREQR387
C                                                                       FREQR388
C                                                                       FREQR389
      IF(FREQ.GT.0.0)GO TO 588                                          FREQR390
C     REDUCE OUT SING SOLN ELEM FOR LOW FREQ MATS                       FREQR391
C                                                                       FREQR392
      M1=1                                                              FREQR393
      M2=0                                                              FREQR394
      IF(IND.EQ.1)GO TO 575                                             FREQR395
      M1=11                                                             FREQR396
      M2=0                                                              FREQR397
  575 I=AMODNO(M1)-M                                                    FREQR398
      IF(I.LE.0)GO TO 580                                               FREQR399
      F(1,I)=0.                                                         FREQR400
      F(2,I)=0.                                                         FREQR401
      DO 576 J=1,NM                                                     FREQR402
      LOC=I+(J-1)*NM                                                    FREQR403
      D(1,LOC)=0.                                                       FREQR404
      D(2,LOC)=0.                                                       FREQR405
      LOC=J+(I-1)*NM                                                    FREQR406
      D(1,LOC)=0.                                                       FREQR407
      D(2,LOC)=0.                                                       FREQR408
  576 CONTINUE                                                          FREQR409
      LOC=I+(I-1)*NM                                                    FREQR410
      D(1,LOC)=1.0                                                      FREQR411
  580 CONTINUE                                                          FREQR412
      IF(IND.EQ.3)GO TO 582                                             FREQR413
      IF(IND.EQ.1.AND.M2.GT.0)GO TO 585                                 FREQR414
      M1=3                                                              FREQR415
      M2=1                                                              FREQR416
      GO TO 575                                                         FREQR417
```

```
  582 IF(M2.GT.0)GO TO 585
      M1=13
      M2=1
      GO TO 575
C
  585 CONTINUE
C
  588 CONTINUE
C
C
      IF(KPRCHK.EQ.0)GO TO 570
      WRITE(NOUT,60)IVBW,FREQ
      CALL PRNT(D,NM,NM,NM,2,4H Z   ,1)
      CALL PRNT(F,NM,1,NM,2,4H F   ,1)
      LINES = LINES+1
  570 CONTINUE
C
C
      DETDP = 0
C
C
C
C
C CALC Z-1 FOR GUST OR FORCE DRIVEN SYSTEM
C
C
      CALL MIS2(D,NM,NM,F,1,NER,DETDP)
C
  590 CONTINUE
C
      M1 = M+1
      M2 = M+NM
      N = 0
      DO 600 I=M1,M2
      N = N+1
      Q(1,I)=F(1,N)
  600 Q(2,I)=F(2,N)
      IF(IND.EQ.3)GO TO 601
      IMOS=IMO
      IRECGS=IRECG
      GO TO 602
  601 IMOA=IMO
      IRECGA=IRECG
  602 CONTINUE
C
      IF (NASYM.EQ.0) GO TO 700
      IF (IND.EQ.3) GO TO 700
      NM = NASYM
      M = NSYM
      IND = 3
      N1 = N1A
      N2 = N2A
```

```
      N3 = N3A
      N4 = N4A
      N5 = N5A
      IMO = IMOA
      IRECG=IRECGA
      GO TO 300
C
  700 CONTINUE
C
C      SYMMETRIC AND ANTI-SYMMETRIC ANALYSES DONE
C
C
      LINES=LINES+1
      IF(IPREQ.EQ.0)GO TO 820
      IF (LINES.LE.KROW) GO TO 880
      CALL HEADNG
      WRITE(NOUT,20)IG,(CR(L,IG),L=1,3)
      WRITE(NOUT,50)VKEAS,VELUN(IUNIT),DYNP,VWG,ALT
      LINES = LINES+7
  880 CONTINUE
      WRITE(NOUT,40)IVBW,VOBW,FREQ
      DO 890 JJ=1,NMD
      LINES=LINES+1
      IF (LINES.LE.KROW) GO TO 895
      CALL HEADNG
      WRITE(NOUT,20)IG,(CR(L,IG),L=1,3)
      WRITE(NOUT,50)VKEAS,VELUN(IUNIT),DYNP,VWG,ALT
      LINES = LINES+7
 895  CONTINUE
      WRITE(NOUT,30)JJ,Q(1,JJ),Q(2,JJ)
  890 CONTINUE
C
      GO TO 850
C
  820 IF(LINES.LE.KROW)GO TO 840
      CALL HEADNG
      WRITE(NOUT,20)IG,(CR(L,IG),L=1,3)
      WRITE(NOUT,50)VKEAS,VELUN(IUNIT),DYNP,VWG,ALT
      LINES=LINES+7
  840 CONTINUE
      WRITE(NOUT,45)IVBW,VOBW,FREQ
C
  850 CONTINUE
C
      IHD(1)=IG
      IHD(2)=IVBW
      IHD(4)=IZROS
      IHD(5)=IZROA
      CALL WRFRSP(4,IG,IVBW,IHD,NOUT,NER)
      CALL WRFRSP (5,IG,IVBW,COEF,NOUT,NER)
      CALL WRFRSP (6,IG,IVBW,IVBWI,NOUT,NER)
      CALL WRFRSP(7,IG,IVBW,Q,NOUT,NER)
```

```
      CALL  WRFRSP(8,IG,IVBW,A(N5S),NOUT,NER)                       FREQR522
      CALL  WRFRSP(9,IG,IVBW,A(N5A),NOUT,NER)                       FREQR523
C                                                                   FREQR524
  800 CONTINUE                                                      FREQR525
C                                                                   FREQR526
C     THIS ENDS FREQ LOOP                                           FREQR527
      IF(ICACS.NE.0)WRITE(NOUT,10)                                  FREQR528
C                                                                   FREQR529
  900 CONTINUE                                                      FREQR530
C                                                                   FREQR531
C     THIS ENDS GUST LOOP                                           FREQR532
C                                                                   FREQR533
C                                                                   FREQR534
      IPREQ=ITEMPQ                                                  FREQR535
      KPRCHK=ITEMPK                                                 FREQR536
      RETURN                                                        FREQR537
C                                                                   FREQR538
 1090 CONTINUE                                                      FREQR539
      WRITE(NOUT,100)                                               FREQR540
      STOP                                                          FREQR541
C                                                                   FREQR542
      END                                                           FREQR543
```

```
      SUBROUTINE GSTINI(IZRO,FGUST,
     1 SPLHP,NMODE,COEF,
     2 F,FI,NM,NK,RS,TANS,IRECG,IPREQ,J)
C
      COMMON/ZZZ/CASE(48),NIN,NOUT,KROW,LINES,IPRNT,NER
C
      DIMENSION F(2,1)
      DIMENSION FI(2,NM,1)
      DIMENSION COEF(1)
      DIMENSION RS(NM,1),TANS(NM,1)
      DIMENSION FGUST(2,1)
      DIMENSION SPLHP(NM,1)
C
C     THIS SUBROUTINE USES THE TRANSPOSE OF THE ABOVE MATRICES
C
      DATA EP/1.0E-02/
C
C     NM = TOTAL AERO ELEMENTS = NBOX+2*NSBETO
C
C
   10 FORMAT(1H0,20X, GUST INTERPOLATION ELEMENTS /1H0,
     1  ELEM   NG   NK ,12X, RR ,12X, RI ,11X, MOD ,11X, TAN )
    9 FORMAT(1H0,20X, SPLH MATRICES )
   20 FORMAT(1H ,3I4,4E14.6)
C
      DO 45 I=1,NMODE
      F(1,I)=0.
   45 F(2,I)=0.
C
      K=0
C
      DO 40 I=1,NK
      IF(COEF(I).EQ.1.0)K=I
   40 CONTINUE
C
      IF(K.NE.0)GO TO 100
C
C
C
      IF(IRECG.NE.0)GO TO 120
C
      ICNT=0
C     SET UP COUNT FOR ZERO GUST LOADS
C
      IF(IPREQ.EQ.0)GO TO 47
C
      CALL HEADNG
      WRITE(NOUT,9)
      CALL PRNT(SPLHP,NM,NMODE,NM,1,4HSPLH,1)
      CALL HEADNG
      WRITE(NOUT,10)
C
```

```
   47 CONTINUE
C
      DO 90 I=1,NM
      DO 80 K=1,NK
      RR=FI(1,I,K)
      RI=FI(2,I,K)
      IF(ABS(RR).LT.EP)RR=0.
      IF(ABS(RI).LT.EP)RI=0.
      RT=SQRT(RR**2+RI**2)
      RS(I,K)=RR
      TANS(I,K)=RI
      IF(IPREQ.NE.0)WRITE(NOUT,20)I,J,K,RR,RI,RT
      IF(RR.NE.0)GO TO 80
      IF(RI.NE.0)GO TO 80
      IF(K.GE.2)GO TO 85
      GO TO 80
C
   80 CONTINUE
C
      GO TO 90
   85 CONTINUE
      DO 86 K=1,NK
      TANS(I,K)=0.
   86 RS(I,K)=0.
      ICNT=ICNT+NK
C
C
   90 CONTINUE
      IRFCG=1
C
      IF(ICNT.GE.NK*NM)IZRO=1
C
      IF(IZRO.EQ.1)GO TO 150
      GO TO 120
C
  100 CONTINUE
      DO 110 I=1,NM
C
      FREAL=FI(1,I,K)
      FIMAG=FI(2,I,K)
      FGUST(1,I)=FREAL
      FGUST(2,I)=FIMAG
C
      DO 182 L=1,NMODE
      F(1,L)=F(1,L)+SPLHP  (I,L )*FREAL
      F(2,L)=F(2,L)+SPLHP  (I,L )*FIMAG
  182 CONTINUE
C
  110 CONTINUE
C
      GO TO 150
C
```

```
  120 CONTINUE                                                          GSTIN106
      DO 140 I=1,NM                                                     GSTIN107
      R=0.                                                              GSTIN108
      THETA=0.                                                          GSTIN109
      DO 130 K=1,NK                                                     GSTIN110
      R=R+COEF(K)*RS(I,K)                                               GSTIN111
  130 THETA=THETA+COEF(K)*TANS(I,K)                                     GSTIN112
C                                                                       GSTIN113
      FREAL=R                                                           GSTIN114
      FIMAG=THETA                                                       GSTIN115
      FGUST(1,I)=FREAL                                                  GSTIN116
      FGUST(2,I)=FIMAG                                                  GSTIN117
C                                                                       GSTIN118
      DO 282 L=1,NMODE                                                  GSTIN119
      F(1,L)=F(1,L)+SPLHP (I,L )*FREAL                                  GSTIN120
      F(2,L)=F(2,L)+SPLHP (I,L )*FIMAG                                  GSTIN121
 282  CONTINUE                                                          GSTIN122
C                                                                       GSTIN123
  140 CONTINUE                                                          GSTIN124
C                                                                       GSTIN125
  150 CONTINUE                                                          GSTIN126
C                                                                       GSTIN127
      IF(IPREQ.EQ.0)GO TO 160                                           GSTIN128
      CALL PRNT(F,NMODE,1,NMODE,2,24HINTERP. GEN. GUST FORCE   ,6)      GSTIN129
C                                                                       GSTIN130
  160 CONTINUE                                                          GSTIN131
C                                                                       GSTIN132
      RETURN                                                            GSTIN133
C                                                                       GSTIN134
      END                                                               GSTIN135
```

```
      SUBROUTINE INTERP(
     1 D,VOBW,DALL,VOBWI,NM,NK,COEF,IPRN,IRECC,INDI,RK,C)
      COMPLEX D(NM,NM)
      COMPLEX DALL(NM,NM,1)
      DIMENSION VOBWI(1)
      DIMENSION COEF(1)
      DIMENSION RK(10),C(13,10)
C
      DATA EPRV/0.010/
      DATA EPC/0.10E-05/
C
C
C     IF(INDI IS SET 0, USE AVAILABLE COEFS
C
      IF(INDI.EQ.0)GO TO 15
C
      II=0
      DO 10 I=1,NK
      COEF(I)=0.
      IF(ABS(1.-VOBWI(I)/VOBW).LE.EPRV)II=I
   10 RK(I)=1.0/VOBWI(I)
      IF(IRECC.EQ.0)CALL FORMC(RK,NK,C,IPRN)
      RKIN=1.0/VOBW
      IF(II.NE.0)GO TO 30
      CALL COEFF(RKIN,RK,NK,C,COEF)
      DO 14 L=1,NK
   14 IF(ABS(COEF(L)).LT.EPC)COEF(L)=0.0
   15 CONTINUE
      DO 20 I=1,NM
      DO 20 J=1,NM
      D(I,J)=0.
      DO 20 L=1,NK
   20 D(I,J)=COEF(L)*DALL(I,J,L)+D(I,J)
      GO TO 50
   30 CONTINUE
      DO 40 I=1,NM
      DO 40 J=1,NM
   40 D(I,J)=DALL(I,J,II)
      COEF(II)=1.0
   50 CONTINUE
      IF(IPRN.EQ.0)GO TO 100
      CALL PRNT(COEF,1,NK,1,1,4HCOEF ,1)
      CALL PRNT (D,NM,NM,NM,2,20HDBAR(INTERPOLATED)   ,5)
  100 RETURN
      END
```

```
      SUBROUTINE TDTINT (VOBW,ISETA,IMO,D,DALL                          TDTINT 2
     1 , FALL,NTOTAP,IG,COEF,NM,NK,KPRCHK,IND,                          TDTINT 3
     3 VOBWI,IVBWI,IRECG,RK,C)                                          TDTINT 4
C                                                                       TDTINT 5
      COMMON /ZZZ/CASE(49),NOUT,KROW,LINES                              TDTINT 6
      COMMON/DISK2/ND2,ITBL2(843),NRECSA,IBUMP,NKD,VOBWS(20)            DISK2  2
C                                                                       TDTINT 8
      DIMENSION D(1)                                                    TDTINT 9
      COMPLEX DALL(NM,NM,1)                                             TDTINT10
      COMPLEX FALL(NTOTAP,NK)                                           TDTINT11
      DIMENSION COEF(1),VOBWI(1),IVBWI(1)                               TDTINT12
      DIMENSION RK(10),C(13,10)                                         TDTINT13
C                                                                       TDTINT14
      M=1                                                               TDTINT15
      IF(IND.EQ.3)M=2                                                   TDTINT16
      NWD = 2*NM*NM                                                     TDTINT17
C                                                                       TDTINT18
C     FIND OUT IF WE NEED NEW AERO                                      TDTINT19
      IRECC=0                                                           TDTINT20
      TESTV=1.E06                                                       TDTINT21
      DO 10 I=1,NKD                                                     TDTINT22
      TEST=ABS(VOBWS(I)-VOBW)                                           TDTINT23
      IF(TEST.GT.TESTV)GO TO 10                                         TDTINT24
      TESTV=TEST                                                        TDTINT25
      IM=I                                                              TDTINT26
   10 CONTINUE                                                          TDTINT27
C                                                                       TDTINT28
      IL=IM-ISETA                                                       TDTINT29
      IF(NK.GT.4)IL=IL-ISETA                                            TDTINT30
      IH=IL+(NK-1)*ISETA                                                TDTINT31
C                                                                       TDTINT32
   20 IF(IL.GE.1.AND.ISETA.GT.0)GO TO 30                                TDTINT33
      IF  (IL.LE.NKD.AND.ISETA.LT.0) GO TO 30                           TDTINT34
      IH=IH+ISETA                                                       TDTINT35
      IM=IM+ISETA                                                       TDTINT36
      IL=IL+ISETA                                                       TDTINT37
      GO TO 20                                                          TDTINT38
   30 IF  (IH.LE.NKD.AND.ISETA.GT.0) GO TO 40                           TDTINT39
      IF(IH.GE.1.AND.ISETA.LT.0)GO TO 40                                TDTINT40
      IH=IH-ISETA                                                       TDTINT41
      IM=IM-ISETA                                                       TDTINT42
      IL=IL-ISETA                                                       TDTINT43
      GO TO 20                                                          TDTINT44
C                                                                       TDTINT45
   40 CONTINUE                                                          TDTINT46
      IF(IM.EQ.IMO)IRECC=1                                              TDTINT47
      IF  (IM.EQ.IMO) GO TO 60                                          TDTINT48
      IRECG = 0                                                         TDTINT49
      N = IL                                                            TDTINT50
      DO 45 I=1,NK                                                      TDTINT51
      IVBWI(I) = N                                                      TDTINT52
      VOBWI(I) = VOBWS(N)                                               TDTINT53
```

```
      LOC=M+(N-1)*IBUMP                                                 TDTINT54
      CALL READMS(ND2,DALL(1,1,I),NWD,LOC)                              TDTINT55
      IF (KPRCHK.EQ.0) GO TO 45                                         TDTINT56
      CALL PRNT (DALL(1,1,I),NM,NM,NM,2,16HINPUT GEN. AERO ,4)          TDTINT57
   45 N = N+ISETA                                                       TDTINT58
C                                                                       TDTINT59
   60 CONTINUE                                                          TDTINT60
C                                                                       TDTINT61
      IF(IRECG.NE.0)GO TO 70                                            TDTINT62
      NWF=2*NTOTAP                                                      TDTINT63
      LOCO=2*(IG-1)+3                                                   TDTINT64
      IF(IND.EQ.3)LOCO=LOCO+1                                           TDTINT65
C                                                                       TDTINT66
      DO 55 K=1,NK                                                      TDTINT67
      N=IVBWI(K)                                                        TDTINT68
      LOC=LOCO+(N-1)*IBUMP                                              TDTINT69
      CALL READMS(ND2,FALL(1,K),NWF,LOC)                                TDTINT70
C                                                                       TDTINT71
   55 CONTINUE                                                          TDTINT72
C                                                                       TDTINT73
   70 CONTINUE                                                          TDTINT74
C                                                                       TDTINT75
      IF (KPRCHK.EQ.0) GO TO 90                                         TDTINT76
      WRITE (NOUT,100) (IVBWI(K),VOBWI(K),K=1,NK)                       TDTINT77
      LINES=LINES+3                                                     TDTINT78
C                                                                       TDTINT79
   90 CONTINUE                                                          TDTINT80
C                                                                       TDTINT81
      INDI=1                                                            TDTINT82
      IF(IND.EQ.3.AND.IRECC.NE.0)INDI=0                                 TDTINT83
C                                                                       TDTINT84
      CALL INTERP(                                                      TDTINT85
     1 D,VOBW,DALL,VOBWI,NM,NK,COEF,KPRCHK,IRECC,INDI,RK,C)             TDTINT86
C                                                                       TDTINT87
      IMO=IM                                                            TDTINT88
C                                                                       TDTINT89
  100 FORMAT(1H0,20X, INTERPOLATING FROM THE FOLLOWING IVBW AND VOBW /  TDTINT90
     1 1H , 10(1X,I3,1X,F8.4))                                          TDTINT91
C                                                                       TDTINT92
      RETURN                                                            TDTINT93
      END                                                               TDTINT94
```

```
      SUBROUTINE TFMATF(NTF,NM,TEMP,D,OMEGA,TFC,TMS,NO,MXORDN,MXORDD,
     1 PHIS,IPRNT)
C
      COMMON/ZZZ/CASE(48),NIN,NOUT,KROW,LINES
C
      DIMENSION TEMP(2,NM,NM),D(2,NM,NM),TFC(2,NTF,1),TMS(NTF,4)
      DIMENSION PHIS(NTF,1)
C
C
      DO 500 N=1,NTF
      M=TMS(N,4)-NO
      DCS=TMS(N,3)
C
      CALL TRFCMX(N,TFC,OMEGA,TFR,TFI,NTF,MXORDN,MXORDD)
C
      DO 200 I=1,NM
      AR=(D(1,I,M)*TFR-D(2,I,M)*TFI)*DCS
      AI=(D(1,I,M)*TFI+D(2,I,M)*TFR)*DCS
      DO 200 J=1,NM
      TEMP(1,I,J)=AR*PHIS(N,J)
      TEMP(2,I,J)=AI*PHIS(N,J)
  200 CONTINUE
C
      IF(IPRNT.EQ.0)GO TO 250
      WRITE(NOUT,1)N,TFR,TFI
    1 FORMAT(1H0,20X, DELT AERO FOR ACS TFN ,I3,5X, TFR= E12.5,5X,
     1  TFI= ,F12.5)
      LINES=LINES+2
      CALL PRNT(TEMP,NM,NM,NM,2,8HACS AERO,2)
  250 CONTINUE
C
      DO 300 I=1,NM
      DO 300 J=1,NM
      D(1,I,J)=D(1,I,J)+TEMP(1,I,J)
  300 D(2,I,J)=D(2,I,J)+TEMP(2,I,J)
C
  500 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE TFMATM(NTF,NM,TEMP,D,OMEGA,TFC,TMS,NO,MXORDN,MXORDD,   TFMATM 2
     1 PHIS,EMBAR,NMD,AMULT,IPRNT)                                      TFMATM 3
C                                                                       TFMATM 4
      COMMON/777/CASE(48),NIN,NOUT,KROW,LINES                           TFMATM 5
C                                                                       TFMATM 6
      DIMENSION TEMP(2,NM,NM),D(2,NM,NM),TFC(2,NTF,1),TMS(NTF,4)        TFMATM 7
      DIMENSION PHIS(NTF,1)                                             TFMATM 8
      DIMENSION EMBAR(NMD,NMD)                                          TFMATM 9
C                                                                       TFMATM10
C                                                                       TFMATM11
      DO 500 N=1,NTF                                                    TFMATM12
      M=TMS(N,4)                                                        TFMATM13
      DCS=TMS(N,3)                                                      TFMATM14
C                                                                       TFMATM15
      CALL TRFCMX(N,TFC,OMEGA,TFR,TFI,NTF,MXORDN,MXORDD)                TFMATM16
C                                                                       TFMATM17
      DO 200 I=1,NM                                                     TFMATM18
      II=I+NO                                                           TFMATM19
      AR=EMBAR(II,M)*TFR*AMULT*DCS                                      TFMATM20
      AI=EMBAR(II,M)*TFI*AMULT*DCS                                      TFMATM21
      DO 200 J=1,NM                                                     TFMATM22
      TEMP(1,I,J)=AR*PHIS(N,J)                                          TFMATM23
      TEMP(2,I,J)=AI*PHIS(N,J)                                          TFMATM24
  200 CONTINUE                                                          TFMATM25
C                                                                       TFMATM26
      IF(IPRNT.EQ.0)GO TO 250                                           TFMATM27
      WRITE(NOUT,1)N,TFR,TFI                                            TFMATM28
    1 FORMAT(1H0,20X, DELT MASS FOR ACS TFN ,I3,5X, TFR= E12.5,5X,      TFMATM29
     1  TFI= ,E12.5)                                                    TFMATM30
      LINES=LINES+2                                                     TFMATM31
      CALL PRNT(TEMP,NM,NM,NM,2,8HACS MASS,2)                           TFMATM32
  250 CONTINUE                                                          TFMATM33
C                                                                       TFMATM34
      DO 300 I=1,NM                                                     TFMATM35
      DO 300 J=1,NM                                                     TFMATM36
      D(1,I,J)=D(1,I,J)+TEMP(1,I,J)                                     TFMATM37
  300 D(2,I,J)=D(2,I,J)+TEMP(2,I,J)                                     TFMATM38
C                                                                       TFMATM39
  500 CONTINUE                                                          TFMATM40
C                                                                       TFMATM41
      RETURN                                                            TFMATM42
      END                                                               TFMATM43
```

```
      SUBROUTINE TRFCMX(N,TFC,OMEGA,TFR,TFI,NTF,MXORDN,MXORDD)
C
C
      DIMENSION TFC(2,NTF,1)
      DOUBLE PRECISION AR,BR,AI,BI,POWR,POWI
C
      AR=0.
      AI=0.
      BR=1.0
      BI=0.
      IF(MXORDN.EQ.0)GO TO 60
      DO 30 I=1,MXORDN,2
      NORD=I-1
      POWR=OMEGA**NORD
      POWI=OMEGA**I
      AR=AR+TFC(1,N,I)*POWR
      AI=AI+TFC(1,N,I+1)*POWI
   30 CONTINUE
C
      IF(MXORDD.EQ.0)GO TO 60
      BR=0.
      DO 50 I=1,MXORDD,2
      NORD=I-1
      POWR=OMEGA**NORD
      POWI=OMEGA**I
      BR=BR+TFC(2,N,I)*POWR
      BI=BI+TFC(2,N,I+1)*POWI
   50 CONTINUE
C
   60 CONTINUE
C
      DEN=BR**2+BI**2
      TFR=(AR*BR+AI*BI)/DEN
      TFI=(AI*BR-AR*BI)/DEN
C
      RETURN
      END
```

```
      SUBROUTINE CFRLOD(CR,                                             CFRLOD 2
     1 PHIX,PHIY,PHIZ,NMS,                                              CFRLOD 3
     1 TMSS,TMSA,TFCS,TFCA,PHISS,PHISA)                                 CFRLOD 4
      COMMON NAA,A(1)                                                   CFRLOD 5
C                                                                       CFRLOD 6
      COMMON /ZZZ/HEDP(48),NIN,NOUT,KROW,LINES, IPRNT,NER               CFRLOD 7
C                                                                       CFRLOD 8
      COMMON /XTF/XF(100)                                               CFRLOD 9
      EQUIVALENCE (XF(6),NSYM),      (XF(7),NASYM),     (XF(8),NM)     CFRLOD10
     1       ,      (XF(9),NFREQ)                                       CFRLOD11
     1       ,      (XF(13),RHOO),     (XF(14),SIGMA),    (XF(18),VTFPS)CFRLOD12
     2 , (XF(29),NACC)                                                  CFRLOD13
      EQUIVALENCE (XF(31),NBEAMS),   (XF(32),NINTLD), (XF(33),NSTRSS)  CFRLOD14
     1       ,      (XF(34),NMGRP),    (XF(35),NABGRP), (XF(36),NSBGRP)CFRLOD15
     2       ,      (XF(37),NBOXES),   (XF(38),NAERSB)                  CFRLOD16
     3       ,      (XF(41),NK),       (XF(42),NG)                      CFRLOD17
     4       ,      (XF(43),NBOX),     (XF(44),NSBETO)                  CFRLOD18
     5 , (XF(71),MXORSN),(XF(72),MXORSD),(XF(73),MXORAN)                CFRLOD19
     6 , (XF(74),MXORAD),(XF(75),NTFS),(XF(76),NTFA)                    CFRLOD20
C                                                                       CFRLOD21
      DIMENSION PHIX(1),PHIY(1),PHIZ(1)                                 CFRLOD22
      DIMENSION TMSS(1),TMSA(1),TFCS(1),TFCA(1),PHISS(1),PHISA(1)       CFRLOD23
      DIMENSION IHD(50)                                                 CFRLOD24
      DIMENSION CR(3,1)                                                 CFRLOD25
      DATA CORERQ/8HUNITGUST/                                           CFRLOD26
C                                                                       CFRLOD27
C                                                                       CFRLOD28
C     READ IN FRSP HEADER                                               CFRLOD29
C                                                                       CFRLOD30
      CALL RDFRSP(1,0,0,IHD,NOUT,NER)                                   CFRLOD31
C                                                                       CFRLOD32
      NTOTAP=IHD(8)                                                     CFRLOD33
C                                                                       CFRLOD34
      L60 = NAA                                                         CFRLOD35
      L61 = L60 + NFREQ                                                 CFRLOD36
      L1  = L61 + NFREQ                                                 CFRLOD37
      L2   =  L1   + 2*MAXO(NINTLD,NACC)*NFREQ                          CFRLOD38
      L3   =  L2   + 2*MAXO(NINTLD,NACC)*NFREQ                          CFRLOD39
      L3A  =  L3   + NFREQ                                              CFRLOD40
      L4   =  L3A  + 2*NACC                                             CFRLOD41
      L5   =  L4   + NK*NFREQ                                           CFRLOD42
      L6   =  L5   + 2*NM*NFREQ                                         CFRLOD43
      L7   =  L6                                                        CFRLOD44
      L8   =  L7   + 2*NINTLD*NSYM                                      CFRLOD45
      LTOT1 = L8 + 2*NINTLD*NASYM                                       CFRLOD46
      L51 = L4                                                          CFRLOD47
      L52 = L51 + NINTLD                                                CFRLOD48
      L9 = L52 + NINTLD                                                 CFRLOD49
      L10 = L9 + 3*NABGRP                                               CFRLOD50
      L11 = L10 + 3*NSBGRP                                              CFRLOD51
      L12 = L11 + NINTLD*NABGRP*NBOXES                                  CFRLOD52
      L13 = L12 + NINTLD*NSBGRP*NAERSB                                  CFRLOD53
```

```
      L62 = L13 + NINTLD*NSBGRP*NAERSB                                  CFRLOD54
      L63 = L62                                                         CFRLOD55
      LTOT2 = L62 + 2*NTOTAP                                            CFRLOD56
C                                                                       CFRLOD57
      LTOT = MAX0(LTOT1,LTOT2)                                          CFRLOD58
      WRITE(NOUT,999)LTOT,CORERQ                                        CFRLOD59
  999 FORMAT(1H0,20X,I10,  WORDS OF CORE RQD FOR STEP +++ ,A10, +++ )   CFRLOD60
      CALL FRLOAD(                                                      CFRLOD61
     1 CP,A(L1),A(L2),A(L3),A(L5),A(L6),A(L7),A(L8)                     CFRLOD62
     1 , A(L9),A(L10),A(L11),A(L12),A(L13),A(L62),A(L63),A(L4),IHD      CFRLOD63
     1 , NTOTAP                                                         CFRLOD64
     1 , PHIX,PHIY,PHIZ,A(L3A),NMS,A(L1),A(L2),NACC                     CFRLOD65
     3 , NINTLD,NM,NSYM,NASYM,NK,NFREQ,NABGRP,NSBGRP,NBOX,NSBETD        CFRLOD66
     4 , NPCXFS,NAERSB,NAEROP,A(L51),A(L52)                             CFRLOD67
     5 ,SIGMA,RHOO,VTFPS,NG                                             CFRLOD68
     6 , NTFS,NTFA,TFCS,TFCA,TMSS,TMSA,PHISS,PHISA,MXORSN,MXORAN,       CFRLOD69
     7 MXORSD,MXORAD,A(L60),A(L61))                                     CFRLOD70
      RETURN                                                            CFRLOD71
      END                                                               CFRLOD72
```

```
      SUBROUTINE FGMULT(K,                                              FGMULT 2
     1 P,NFREQ,VWG,NFLOC,FGL,PINT,NINTLD,NGRP,NFNL,NTOTAP,SCODE)        FGMULT 3
C                                                                       FGMULT 4
      COMMON/777/HEDR(48),NIN,NOUT,KROW,LINES                           FGMULT 5
C                                                                       FGMULT 6
      COMMON/XTF/NF(100)                                                FGMULT 7
      EQUIVALENCE (NF(20),KPRCHK)                                       FGMULT 8
C                                                                       FGMULT 9
      DIMENSION P(2,NINTLD,NFREQ)                                       FGMULT10
      DIMENSION NFNL(3,NGRP)                                            FGMULT11
      DIMENSION PINT(NINTLD,NGRP,1)                                     FGMULT12
      DIMENSION FGL(2,NTOTAP)                                           FGMULT13
      DIMENSION SCODE(1)                                                FGMULT14
C                                                                       FGMULT15
      DO 150 L=1,NGRP                                                   FGMULT16
      N=NFNL(2,L)-NFNL(1,L)+1                                           FGMULT17
      DO 150 J=1,N                                                      FGMULT18
      LOC=NFLOC+NFNL(1,L)+J-1                                           FGMULT19
      GR=FGL(1,LOC)                                                     FGMULT20
      GI=FGL(2,LOC)                                                     FGMULT21
C                                                                       FGMULT22
      DO 150 M=1,NINTLD                                                 FGMULT23
      P(1,M,K)=P(1,M,K)+GR*VWG*PINT(M,L,J)*SCODE(M)                     FGMULT24
  150 P(2,M,K)=P(2,M,K)+GI*VWG*PINT(M,L,J)*SCODE(M)                     FGMULT25
C                                                                       FGMULT26
      RETURN                                                            FGMULT27
      END                                                               FGMULT28
```

```
      SUBROUTINE FRLOAD(CR,
     1 PS,PA,OMEGA,Q,PIQ,PAQS,PAQA,NFNLAB,NFNLSB,PINTP,PINTZ,PINTY,
     1 FGPS,FGPA,COFF,IHD,NTOTAP,
     1 PHIX,PHIY,PHIZ,INDACC,NMS,AS,AA,NACC,
     1 NINTLD,NM,NSYM,NASYM,NK,NFREQ,NABGRP,NSBGRP,VBOX,NSBETO,
     1 NBOXES,NAERSB,NAEROP,SYMCOD,ASMCOD,
     1 SIGMA,RHO,VTFPS,NG,
     1 NTFS,NTFA,TFCS,TFCA,TMSS,TMSA,PHISS,PHISA,
     1 MXORSN,MXORAN,MXORSD,MXORAC,PLOTF,PLOTV)
C
      COMMON /ZZZ/HEDP(42),DT(6),NIN,NOUT,KROW,LINES,IPRNT,NER
C
      COMMON/XTF/NF(100)
C
      EQUIVALENCE (NF(20),KPRCHK),(NF(21),KPRCXI)
     1 , (NF(100),SIZFCT)
     2 , (NF(26),IPLQ)
     3 ,   (NF(27),IPLL)
C
C
      DIMENSION OMEGA(NFREQ)
      DIMENSION PS(2,NINTLD,NFREQ)
      DIMENSION PA(2,NINTLD,NFREQ)
      DIMENSION COFF(NK,NFREQ)
      DIMENSION PIQ(NINTLD,NM)
      DIMENSION Q(2,NM,NFREQ)
      DIMENSION PAQS(2,NINTLD,NSYM)
      DIMENSION PAQA(2,NINTLD,NASYM)
      DIMENSION NFNLAB(3,NABGRP)
      DIMENSION NFNLSB(3,NSBGRP)
      DIMENSION PINTP(NINTLD,NABGRP,NBOXES)
      DIMENSION PINTZ(NINTLD,NSBGRP,NAERSB)
      DIMENSION PINTY(NINTLD,NSBGRP,NAERSB)
      DIMENSION FGPS(2,NTOTAP)
      DIMENSION FGPA(2,NTOTAP)
      DIMENSION PHIX(NMS,1),PHIY(NMS,1),PHIZ(NMS,1),INDACC(2,NACC)
      DIMENSION AS(2,NACC,NFREQ),  AA(2,NACC,NFREQ)
      DIMENSION TFCS(2,NTFS,1),TMSS(NTFS,4),PHISS(NTFS,NSYM)
      DIMENSION TFCA(2,NTFA,1),TMSA(NTFA,4),PHISA(NTFA,NASYM)
      DIMENSION SYMCOD(1),ASMCOD(1)
      DIMENSION PLOTF(1),PLOTV(1)
      DIMENSION COEFT(10),IVBWT(10)
C
      DIMENSION IHD(50)
      DIMENSION CR(3,1)
C
      DATA TWOPI/6.283185/
C
C     KPRCXI=1 PRNTS LOADS VS FREQ  KPRCHK=1 PRNTS STUFF
C
      ITEMPI=KPRCXI
      ITEMPK=KPRCHK
```

```
      IPRI1=ITEMPI                                                      FRLOAD54
      IPRI2=ITEMPI                                                      FRLOAD55
      IPRK1=ITEMPK                                                      FRLOAD56
      IPRK2=ITEMPK                                                      FRLOAD57
      IPRK2=ITEMPK-IPRK1*10                                             FRLOAD58
C                                                                       FRLOAD59
C                                                                       FRLOAD60
      FACTOR=SIZFCT**2                                                  FRLOAD61
C                                                                       FRLOAD62
      IHD(5) = NINTLD                                                   FRLOAD63
      IHD(6) = NACC                                                     FRLOAD64
      DO 10 I=7,10                                                      FRLOAD65
   10 IHD(I) = 0                                                        FRLOAD66
      CALL WRLOAD (1,0,IHD,NOUT,NER)                                    FRLOAD67
C                                                                       FRLOAD68
C     READ STALDS INTO PIQ AND WRITE OUT ON LOAD DATASET                FRLOAD69
      CALL RDUNIT (2,0,PIQ,NOUT,NER)                                    FRLOAD70
      CALL WRLOAD (2,0,PIQ,NOUT,NER)                                    FRLOAD71
C                                                                       FRLOAD72
      CALL RDFRSP (2,0,0,CR,NOUT,NER)                                   FRLOAD73
      CALL WRLOAD (3,0,CR,NOUT,NER)                                     FRLOAD74
C                                                                       FRLOAD75
      CALL RDFRSP (3,0,0,OMEGA,NOUT,NER)                                FRLOAD76
      CALL WRLOAD (4,0,OMEGA,NOUT,NER)                                  FRLOAD77
C                                                                       FRLOAD78
      IF (NACC.EQ.0) GO TO 20                                           FRLOAD79
C                                                                       FRLOAD80
      READ (NIN,15) INDACC                                              FRLOAD81
   15 FORMAT (6I12)                                                     FRLOAD82
      CALL HEADNG                                                       FRLOAD83
      WRITE (NOUT,16) (J,(INDACC(I,J),I=1,2),J=1,NACC)                  FRLOAD84
   16 FORMAT (1H0,5X1HI,2X4HMASS,3X3HDOF/(1X3I6) )                      FRLOAD85
      CALL WRLOAD (5,0,INDACC,NOUT,NER)                                 FRLOAD86
C                                                                       FRLOAD87
   20 CONTINUE                                                          FRLOAD88
C                                                                       FRLOAD89
      DYNP=SIGMA*RHO*VTFPS**2*FACTOR/288.                               FRLOAD90
      VWG=DYNP/VTFPS                                                    FRLOAD91
C                                                                       FRLOAD92
C                                                                       FRLOAD93
      DO 300 IG=1,NG                                                    FRLOAD94
C                                                                       FRLOAD95
      KPRCHK=0                                                          FRLOAD96
      KPRCXI=0                                                          FRLOAD97
      IF(IG.EQ.IPRI1.OR.IG.EQ.IPRI2)KPRCXI=1                            FRLOAD98
      IF(IG.EQ.IPRK1.OR.IG.EQ.IPRK2)KPRCHK=1                            FRLOAD99
      IF(ITEMPI.GT.99)KPRCXI=1                                          FRLOA100
C                                                                       FRLOA101
C     IG IS GUST ORIENTATION INDEX                                      FRLOA102
C                                                                       FRLOA103
C                                                                       FRLOA104
C     READ BACK ZERO CHECKS                                             FRLOA105
```

```
      CALL RDFRSP(4,IG,1,IHD,NCUT,NER)                                  FRLOA106
      IZROS=IHD(4)                                                      FRLOA107
      IZROA=IHD(5)                                                      FRLOA108
C                                                                       FRLOA109
C                                                                       FRLOA110
C     READ IN PIQ FROM NTU                                              FRLOA111
C                                                                       FRLOA112
      CALL RDUNIT (5,0,PIQ,NOUT,NER)                                    FRLOA113
C                                                                       FRLOA114
C     READ IN ALL COEF AND Q FOR THIS IG FROM NTQ                       FRLOA115
C                                                                       FRLOA116
      DO 50 I=1,NK                                                      FRLOA117
      DO 60 J=1,NFREQ                                                   FRLOA118
   60 COEF(I,J)=0.0                                                     FRLOA119
C                                                                       FRLOA120
C                                                                       FRLOA121
      DO 100 K=1,NFREQ                                                  FRLOA122
C     READ COEF                                                         FRLOA123
C     READ Q                                                            FRLOA124
      CALL RDFRSP(5,IG,K,COEFT,NOUT,NER)                                FRLOA125
      CALL RDFRSP(6,IG,K,IVBWT,NOUT,NER)                                FRLOA126
      CALL RDFRSP (7,IG,K,Q(1,1,K),NOUT,NER)                            FRLOA127
      DO 100 J=1,NK                                                     FRLOA128
      JJ=IVBWT(J)                                                       FRLOA129
      COEF(JJ,K)=COEFT(J)                                               FRLOA130
  100 CONTINUE                                                          FRLOA131
C                                                                       FRLOA132
      IF(IPLQ.EQ.0)GO TO 90                                             FRLOA133
      IPLQ1=IPLQ/10                                                     FRLOA134
      IPLQ2=IPLQ-10*IPLQ1                                               FRLOA135
      IF(IPLQ1.NE.IG.AND.IPLQ2.NE.IG)GO TO 90                           FRLOA136
      DO 50 J=1,NM                                                      FRLOA137
      IF(J.LE.NSYM.AND.IZROS.EQ.1)GO TO 50                              FRLOA138
      IF(J.GT.NSYM.AND.IZROA.EQ.1)GO TO 50                              FRLOA139
      DO 40 I=1,NFREQ                                                   FRLOA140
      PLOTF(I)=OMEGA(I)/TWOPI                                           FRLOA141
   40 PLOTV(I)=SQRT(Q(1,J,I)**2+Q(2,J,I)**2)                            FRLOA142
      CALL PLT1(NFREQ,PLOTF,PLOTV,1)                                    FRLOA143
      WRITE(NOUT,5)IG,J                                                 FRLOA144
    5 FORMAT(1H0,20X, Q PLOTS FOR ORIENTATION  ,I4,  MODE NO. ,I4)      FRLOA145
   50 CONTINUE                                                          FRLOA146
   90 CONTINUE                                                          FRLOA147
C                                                                       FRLOA148
      IF(KPRCHK.EQ.0)GO TO 101                                          FRLOA149
      CALL HEADNG                                                       FRLOA150
      WRITE(NOUT,4)IG                                                   FRLOA151
    4 FORMAT(1H ,20X, HERE COMES LOTS OF CHECK PRINT FOR ORIENT ,I4)    FRLOA152
      WRITE(NOUT,9)DYNP,VWG                                             FRLOA153
    9 FORMAT(1H0,20X, Q =    ,F10.4,5X, Q/V =   ,F10.4)                 FRLOA154
      CALL PRNT(OMEGA,NFREQ,1,NFREQ,1,8H OMEGA  ,2)                     FRLOA155
      CALL PRNT(COEF,NK,NFREQ,NK,1,8HINT.COEF ,2)                       FRLOA156
      CALL PRNT(Q,NM,NFREQ,NM,2,4H Q   ,1)                              FRLOA157
```

```
      CALL PRNT(PIQ,NINTLD,NM,NINTLD,1,4H PIQ   ,1)                     FRLOA158
  101 CONTINUE                                                          FRLOA159
C                                                                       FRLOA160
      IF (NACC.EQ.0) GO TO 1102                                         FRLOA161
C                                                                       FRLOA162
C *** CALC -OMEGA**2 X PHI X Q                                          FRLOA163
C                                                                       FRLOA164
      DO 1115 J=1,NFREQ                                                 FRLOA165
      W2 = -OMEGA(J)**2/386.088                                         FRLOA166
C                                                                       FRLOA167
      DO 1115 I=1,NACC                                                  FRLOA168
      AS(1,I,J) = 0.0                                                   FRLOA169
      AS(2,I,J) = 0.0                                                   FRLOA170
      AA(1,I,J) = 0.0                                                   FRLOA171
      AA(2,I,J) = 0.0                                                   FRLOA172
      MP = INDACC(1,I)                                                  FRLOA173
      L = INDACC(2,I)                                                   FRLOA174
      IF (IZROS.EQ.1) GO TO 1109                                        FRLOA175
      DO 1105 K=1,NSYM                                                  FRLOA176
      IF (L.EQ.1) PHI = PHIX(MP,K)                                      FRLOA177
      IF (L.EQ.2) PHI = PHIY(MP,K)                                      FRLOA178
      IF (L.EQ.3) PHI = PHIZ(MP,K)                                      FRLOA179
      AS(1,I,J) = AS(1,I,J) + W2*PHI*Q(1,K,J)                           FRLOA180
 1105 AS(2,I,J) = AS(2,I,J) + W2*PHI*Q(2,K,J)                           FRLOA181
C                                                                       FRLOA182
      IF (NTFS.EQ.0) GO TO 1109                                         FRLOA183
C                                                                       FRLOA184
      DO 1108 N=1,NTFS                                                  FRLOA185
      M = TMSS(N,4)                                                     FRLOA186
      DCS = TMSS(N,3)                                                   FRLOA187
      IF (L.EQ.1) PHI = PHIX(MP,M)                                      FRLOA188
      IF (L.EQ.2) PHI = PHIY(MP,M)                                      FRLOA189
      IF (L.EQ.3) PHI = PHIZ(MP,M)                                      FRLOA190
      IF (PHI.EQ.0.) GO TO 1108                                         FRLOA191
C                                                                       FRLOA192
      CALL TRFCMX (N,TFCS,OMEGA(J),TFR,TFI,NTFS,MXORSN,MXORSD)          FRLOA193
      ARQ = 0.                                                          FRLOA194
      AIQ = 0.                                                          FRLOA195
      DO 1107 K=1,NSYM                                                  FRLOA196
      ARQ = ARQ+PHISS(N,K)*Q(1,K,J)                                     FRLOA197
 1107 AIQ = AIQ+PHISS(N,K)*Q(2,K,J)                                     FRLOA198
      AR = (TFR*ARQ-TFI*AIQ)*DCS                                        FRLOA199
      AI = (TFR*AIQ+TFI*ARQ)*DCS                                        FRLOA200
      AS(1,I,J) = AS(1,I,J) + W2*PHI*AR                                 FRLOA201
      AS(2,I,J) = AS(2,I,J) + W2*PHI*AI                                 FRLOA202
 1108 CONTINUE                                                          FRLOA203
C                                                                       FRLOA204
 1109 IF (NASYM.EQ.0) GO TO 1114                                        FRLOA205
C                                                                       FRLOA206
      IF (IZROA.EQ.1) GO TO 1114                                        FRLOA207
C                                                                       FRLOA208
      DO 1110 KK=1,NASYM                                                FRLOA209
```

```
      K = KK+NSYM                                                       FRLOA210
      IF (L.EQ.1) PHI = PHIX(MP,K)                                      FRLOA211
      IF (L.EQ.2) PHI = PHIY(MP,K)                                      FRLOA212
      IF (L.EQ.3) PHI = PHIZ(MP,K)                                      FRLOA213
      AA(1,I,J) = AA(1,I,J) + W2*PHI*Q(1,K,J)                           FRLOA214
 1110 AA(2,I,J) = AA(2,I,J) + W2*PHI*Q(2,K,J)                           FRLOA215
C                                                                       FRLOA216
      IF (NTFA.EQ.0) GO TO 1114                                         FRLOA217
C                                                                       FRLOA218
      DO 1113 N=1,NTFA                                                  FRLOA219
      M = TMSA(N,4)                                                     FRLOA220
      DCS = TMSA(N,3)                                                   FRLOA221
      IF (L.EQ.1) PHI = PHIX(MP,M)                                      FRLOA222
      IF (L.EQ.2) PHI = PHIY(MP,M)                                      FRLOA223
      IF (L.EQ.3) PHI = PHIZ(MP,M)                                      FRLOA224
      IF (PHI.EQ.0.) GO TO 1113                                         FRLOA225
C                                                                       FRLOA226
      CALL TRFCMX (N,TFCA,OMEGA(J),TFR,TFI,NTFA,MXORAN,MXORAD)          FRLOA227
      ARQ = 0.                                                          FRLOA228
      AIQ = 0.                                                          FRLOA229
      DO 1112 KK=1,NASYM                                                FRLOA230
      K = KK+NSYM                                                       FRLOA231
      ARQ = ARQ+PHISA(N,K)*Q(1,K,J)                                     FRLOA232
 1112 AIQ = AIQ+PHISA(N,K)*Q(2,K,J)                                     FRLOA233
      AR = (TFR*ARQ-TFI*AIQ)*DCS                                        FRLOA234
      AI = (TFR*AIQ+TFI*ARQ)*DCS                                        FRLOA235
      AA(1,I,J) = AA(1,I,J) + W2*PHI*AR                                 FRLOA236
      AA(2,I,J) = AA(2,I,J) + W2*PHI*AI                                 FRLOA237
 1113 CONTINUE                                                          FRLOA238
C                                                                       FRLOA239
 1114 CONTINUE                                                          FRLOA240
 1115 CONTINUE                                                          FRLOA241
C                                                                       FRLOA242
      CALL WRLOAD (6,IG,AS,NOUT,NER)                                    FRLOA243
      CALL WRLOAD (7,IG,AA,NOUT,NER)                                    FRLOA244
C                                                                       FRLOA245
      IF (KPRCHK.EQ.0) GO TO 1102                                       FRLOA246
      LINES = 0                                                         FRLOA247
      IF (IZROS.EQ.0) CALL PRNT                                         FRLOA248
     1 (AS,NACC,NFRFQ,NACC,2,20H SYM ACCELERATIONS  ,5)                 FRLOA249
      IF (NASYM.NE.0.AND.IZROA.EQ.0) CALL PRNT                          FRLOA250
     1 (AA,NACC,NFRFQ,NACC,2,20HASYM ACCELERATIONS  ,5)                 FRLOA251
C                                                                       FRLOA252
 1102 CONTINUE                                                          FRLOA253
C                                                                       FRLOA254
C                                                                       FRLOA255
C     CALC OMEGA**2 X PIQ X Q                                           FRLOA256
C                                                                       FRLOA257
      DO 115 J=1,NFRFQ                                                  FRLOA258
      AMULT=OMEGA(J)**2                                                 FRLOA259
      DO 115 I=1,NINTLD                                                 FRLOA260
      PS(1,I,J)=0.                                                      FRLOA261
```

```
      PS(2,I,J)=0.                                                      FRLOA262
      PA(1,I,J)=0.                                                      FRLOA263
      PA(2,I,J)=0.                                                      FRLOA264
      IF(IZROS.EQ.1)GO TO 109                                           FRLOA265
      DO 105 K=1,NSYM                                                   FRLOA266
      PS(1,I,J)=PS(1,I,J)+PIQ(I,K)*Q(1,K,J)*AMULT                       FRLOA267
  105 PS(2,I,J)=PS(2,I,J)+PIQ(I,K)*Q(2,K,J)*AMULT                       FRLOA268
      IF(NTFS.EQ.0)GO TO 109                                            FRLOA269
      DO 108 N=1,NTFS                                                   FRLOA270
      CALL TRFCMX(N,TFCS,OMEGA(J),TFR,TFI,NTFS,MXORSN,MXORSD)           FRLOA271
      ARQ=0.                                                            FRLOA272
      AIQ=0.                                                            FRLOA273
      M=TMSS(N,4)                                                       FRLOA274
      DCS=TMSS(N,3)                                                     FRLOA275
      DO 107 K=1,NSYM                                                   FRLOA276
      ARQ=ARQ+PHISS(N,K)*Q(1,K,J)                                       FRLOA277
  107 AIQ=AIQ+PHISS(N,K)*Q(2,K,J)                                       FRLOA278
      AR=(TFR*ARQ-TFI*AIQ)*DCS                                          FRLOA279
      AI=(TFR*AIQ+TFI*ARQ)*DCS                                          FRLOA280
      PS(1,I,J)=PS(1,I,J)+PIQ(I,M)*AR*AMULT                             FRLOA281
      PS(2,I,J)=PS(2,I,J)+PIQ(I,M)*AI*AMULT                             FRLOA282
  108 CONTINUE                                                          FRLOA283
  109 CONTINUE                                                          FRLOA284
      IF(NASYM.EQ.0)GO TO 114                                           FRLOA285
      IF(IZROA.EQ.1)GO TO 114                                           FRLOA286
      DO 110 K=1,NASYM                                                  FRLOA287
      KK=K+NSYM                                                         FRLOA288
      PA(1,I,J)=PA(1,I,J)+PIQ(I,KK)*Q(1,KK,J)*AMULT                     FRLOA289
  110 PA(2,I,J)=PA(2,I,J)+PIQ(I,KK)*Q(2,KK,J)*AMULT                     FRLOA290
      IF (NTFA.EQ.0) GO TO 114                                          FRLOA291
      DO 113 N=1,NTFA                                                   FRLOA292
      CALL TRFCMX(N,TFCA,OMEGA(J),TFR,TFI,NTFA,MXORAN,MXORAD)           FRLOA293
      ARQ=0.                                                            FRLOA294
      AIQ=0.                                                            FRLOA295
      M=TMSA(N,4)                                                       FRLOA296
      DCS=TMSA(N,3)                                                     FRLOA297
      DO 112 K=1,NASYM                                                  FRLOA298
      KK=K+NSYM                                                         FRLOA299
      ARQ=ARQ+PHISA(N,K)*Q(1,KK,J)                                      FRLOA300
  112 AIQ=AIQ+PHISA(N,K)*Q(2,KK,J)                                      FRLOA301
      AR=(TFR*ARQ-TFI*AIQ)*DCS                                          FRLOA302
      AI=(TFR*AIQ+TFI*ARQ)*DCS                                          FRLOA303
      PA(1,I,J)=PA(1,I,J)+PIQ(I,M)*AR*AMULT                             FRLOA304
      PA(2,I,J)=PA(2,I,J)+PIQ(I,M)*AI*AMULT                             FRLOA305
  113 CONTINUE                                                          FRLOA306
  114 CONTINUE                                                          FRLOA307
  115 CONTINUE                                                          FRLOA308
C                                                                       FRLOA309
      IF(KPRCHK.EQ.0)GO TO 102                                          FRLOA310
      IF(IZROS.EQ.0)CALL PRNT(                                          FRLOA311
     1 PS,NINTLD,NFREQ,NINTLD,2,20H SYM INERTIAL LOADS    ,5)           FRLOA312
      IF(NASYM.NE.0.AND.IZROA.EQ.0)CALL PRNT(                           FRLOA313
```

```
     1 PA,NINTLD,NFREQ,NINTLD,2,20HASYM INERTIAL LOADS    ,5)       FRLOA314
 102 CONTINUE                                                       FRLOA315
C                                                                   FRLOA316
C                                                                   FRLOA317
C     CALC DYNP X MOTION DEPENDENT AERO FORCES X Q                  FRLOA318
C                                                                   FRLOA319
      DO 200 L=1,NK                                                 FRLOA320
C                                                                   FRLOA321
C     READ IN PAQS(L) AND PAQA(L) FOR THIS IG                       FRLOA322
C                                                                   FRLOA323
      IF(IZROS.EQ.1)GO TO 116                                       FRLOA324
C                                                                   FRLOA325
      CALL RDUNIT (7,L,PAQS,NOUT,NER)                               FRLOA326
C                                                                   FRLOA327
 116 CONTINUE                                                       FRLOA328
C                                                                   FRLOA329
      IF(NASYM.EQ.0)GO TO 117                                       FRLOA330
      IF(IZROA.EQ.1)GO TO 117                                       FRLOA331
C                                                                   FRLOA332
      CALL RDUNIT (8,L,PAQA,NOUT,NER)                               FRLOA333
C                                                                   FRLOA334
 117 CONTINUE                                                       FRLOA335
      IF(KPRCHK.EQ.0)GO TO 118                                      FRLOA336
      IF(IZROS.EQ.0)CALL PRNT(                                      FRLOA337
     1 PAQS,NINTLD,NSYM,NINTLD,2,4HPAQS   ,1)                       FRLOA338
      IF(NASYM.NE.0.AND.IZROA.EQ.0)CALL PRNT(                       FRLOA339
     1 PAQA,NINTLD,NASYM,NINTLD,2,4HPAQA   ,1)                      FRLOA340
C                                                                   FRLOA341
 118 CONTINUE                                                       FRLOA342
      DO 140 J=1,NFREQ                                              FRLOA343
      AMULT=DYNP*COEF(L,J)                                          FRLOA344
      DO 140 I=1,NINTLD                                             FRLOA345
C                                                                   FRLOA346
      IF(IZROS.EQ.1)GO TO 125                                       FRLOA347
      DO 120 K=1,NSYM                                               FRLOA348
      PS(1,I,J)=PS(1,I,J)+                                          FRLOA349
     1 (PAQS(1,I,K)*Q(1,K,J)-PAQS(2,I,K)*Q(2,K,J))*AMULT            FRLOA350
 120 PS(2,I,J)=PS(2,I,J)+                                           FRLOA351
     1 (PAQS(1,I,K)*Q(2,K,J)+PAQS(2,I,K)*Q(1,K,J))*AMULT            FRLOA352
      IF(NTFS.EQ.0)GO TO 125                                        FRLOA353
      DO 124 N=1,NTFS                                               FRLOA354
      CALL TRFCMX(N,TFCS,OMEGA(J),TFR,TFI,NTFS,MXORSN,MXORSD)       FRLOA355
      ARQ=0.                                                        FRLOA356
      AIQ=0.                                                        FRLOA357
      M=TMSS(N,4)                                                   FRLOA358
      DCS=TMSS(N,3)                                                 FRLOA359
      DO 123 K=1,NSYM                                               FRLOA360
      ARQ=ARQ+PHISS(N,K)*Q(1,K,J)                                   FRLOA361
 123 AIQ=AIQ+PHISS(N,K)*Q(2,K,J)                                    FRLOA362
      AR=(TFR*ARQ-TFI*AIQ)*DCS                                      FRLOA363
      AI=(TFR*AIQ+TFI*ARQ)*DCS                                      FRLOA364
      PS(1,I,J)=PS(1,I,J)+(PAQS(1,I,M)*AR-PAQS(2,I,M)*AI)*AMULT     FRLOA365
```

```
      PS(2,I,J)=PS(2,I,J)+(PAQS(1,I,M)*AI+PAQS(2,I,M)*AR)*AMULT            FRLOA366
  124 CONTINUE                                                              FRLOA367
  125 CONTINUE                                                              FRLOA368
      IF(NASYM.EQ.0)GO TO 140                                               FRLOA369
      IF(IZROA.EQ.1)GO TO 140                                               FRLOA370
      DO 130 K=1,NASYM                                                      FRLOA371
      KK=K+NSYM                                                             FRLOA372
      PA(1,I,J)=PA(1,I,J)+                                                  FRLOA373
     1 (PAQA(1,I,K)*Q(1,KK,J)-PAQA(2,I,K)*Q(2,KK,J))*AMULT                 FRLOA374
  130 PA(2,I,J)=PA(2,I,J)+                                                  FRLOA375
     1 (PAQA(1,I,K)*Q(2,KK,J)+PAQA(2,I,K)*Q(1,KK,J))*AMULT                 FRLOA376
      IF(NTFA.EQ.0)GO TO 140                                                FRLOA377
      DO 135 N=1,NTFA                                                       FRLOA378
      CALL TRFCMX(N,TFCA,OMEGA(J),TFR,TFI,NTFA,MXORAN,MXORAD)               FRLOA379
      ARQ=0.                                                                FRLOA380
      AIQ=0.                                                                FRLOA381
      M=TMSA(N,4)-NSYM                                                      FRLOA382
      DCS=TMSA(N,3)                                                         FRLOA383
      DO 134 K=1,NASYM                                                      FRLOA384
      KK=K+NSYM                                                             FRLOA385
      ARQ=ARQ+PHISA(N,K)*Q(1,KK,J)                                          FRLOA386
  134 AIQ=AIQ+PHISA(N,K)*Q(2,KK,J)                                          FRLOA387
      AR=(TFR*ARQ-TFI*AIQ)*DCS                                              FRLOA388
      AI=(TFR*AIQ+TFI*ARQ)*DCS                                              FRLOA389
      PA(1,I,J)=PA(1,I,J)+(PAQA(1,I,M)*AR-PAQA(2,I,M)*AI)*AMULT            FRLOA390
      PA(2,I,J)=PA(2,I,J)+(PAQA(1,I,M)*AI+PAQA(2,I,M)*AR)*AMULT            FRLOA391
  135 CONTINUE                                                              FRLOA392
  140 CONTINUE                                                              FRLOA393
C                                                                           FRLOA394
  200 CONTINUE                                                              FRLOA395
C                                                                           FRLOA396
      IF(KPPCHK.EQ.0)GO TO 201                                              FRLOA397
C                                                                           FRLOA398
      IF(IZROS.EQ.0)CALL PRNT(                                              FRLOA399
     1 PS,NINTLD,NFREQ,NINTLD,2,24H SYM INERT+MD AERO LOADS    ,6)         FRLOA400
      IF(NASYM.NE.0.AND.IZROA.EQ.0)CALL PRNT(                               FRLOA401
     1 PA,NINTLD,NFREQ,NINTLD,2,24HASYM INERT+MD AERO LOADS    ,6)         FRLOA402
C                                                                           FRLOA403
  201 CONTINUE                                                              FRLOA404
C                                                                           FRLOA405
C     NOW CALC GUST LOADS                                                   FRLOA406
C                                                                           FRLOA407
C                                                                           FRLOA408
C     READ IN SYMCOD AND ASMCOD                                             FRLOA409
      CALL RDUNIT (9,0,SYMCOD,NOUT,NER)                                     FRLOA410
      CALL RDUNIT (10,0,ASMCOD,NOUT,NER)                                    FRLOA411
C                                                                           FRLOA412
C     GUST LOADS ON PANELS                                                  FRLOA413
C                                                                           FRLOA414
C     READ IN NFNLAB,PINTP                                                  FRLOA415
      CALL RDUNIT (11,0,NFNLAB,NOUT,NER)                                    FRLOA416
      CALL RDUNIT (12,0,PINTP,NOUT,NER)                                     FRLOA417
```

```
C                                                                        FRLOA418
C     READ IN NFNLSB,PINTZ                                               FRLOA419
      CALL RDUNIT(13,0,NFNLSB,NOUT,NER)                                  FRLOA420
      CALL RDUNIT(14,0,PINTZ ,NOUT,NER)                                  FRLOA421
C     READ IN PINTY                                                      FRLOA422
      CALL RDUNIT(15,0,PINTY ,NOUT,NER)                                  FRLOA423
C                                                                        FRLOA424
      DO 260 J=1,NFREQ                                                   FRLOA425
C                                                                        FRLOA426
      IF(IZROS.EQ.1)GO TO 401                                            FRLOA427
C                                                                        FRLOA428
      CALL RDFRSP(8,IG,J,FGPS,NOUT,NER)                                  FRLOA429
      NFLOC=0                                                            FRLOA430
C                                                                        FRLOA431
C     NOW CALC SYMMETRIC INTGD GUST LOADS                                FRLOA432
C                                                                        FRLOA433
      CALL FGMULT(J,                                                     FRLOA434
     1 PS,NFREQ,VWG,NFLOC,FGPS,PINTP,NINTLD,NABGRP,NFNLAB,NTOTAP,        FRLOA435
     2 SYMCOD)                                                           FRLOA436
C                                                                        FRLOA437
      NFLOC=NBOX                                                         FRLOA438
C                                                                        FRLOA439
      CALL FGMULT(J,                                                     FRLOA440
     1 PS,NFREQ,VWG,NFLOC,FGPS,PINTZ,NINTLD,NSBGRP,NFNLSB,NTOTAP,        FRLOA441
     2 SYMCOD)                                                           FRLOA442
C                                                                        FRLOA443
      NFLOC=NBOX+NSBETO                                                  FRLOA444
C                                                                        FRLOA445
      CALL FGMULT(J,                                                     FRLOA446
     1 PS,NFREQ,VWG,NFLOC,FGPS,PINTY,NINTLD,NSBGRP,NFNLSB,NTOTAP,        FRLOA447
     2 SYMCOD)                                                           FRLOA448
C                                                                        FRLOA449
  401 IF(NASYM.EQ.0)GO TO 402                                            FRLOA450
      IF(IZROA.EQ.1)GO TO 402                                            FRLOA451
C                                                                        FRLOA452
      CALL RDFRSP(9,IG,J,FGPA,NOUT,NER)                                  FRLOA453
C                                                                        FRLOA454
      NFLOC=0                                                            FRLOA455
C                                                                        FRLOA456
C     NOW CALC ANTISYMMETRIC INTGD GUST LOADS                            FRLOA457
C                                                                        FRLOA458
      CALL FGMULT(J,                                                     FRLOA459
     1 PA,NFREQ,VWG,NFLOC,FGPA,PINTP,NINTLD,NABGRP,NFNLAB,NTOTAP,        FRLOA460
     2 ASMCOD)                                                           FRLOA461
C                                                                        FRLOA462
      NFLOC=NBOX                                                         FRLOA463
C                                                                        FRLOA464
      CALL FGMULT(J,                                                     FRLOA465
     1 PA,NFREQ,VWG,NFLOC,FGPA,PINTZ,NINTLD,NSBGRP,NFNLSB,NTOTAP,        FRLOA466
     2 ASMCOD)                                                           FRLOA467
C                                                                        FRLOA468
      NFLOC=NBOX+NSBETO                                                  FRLOA469
```

```
C                                                                         FRLOA470
      CALL FGMILT(J,                                                      FRLOA471
     1 PA,NFREQ,VWG,NFLOC,FGPA,PINTY,NINTLD,NSBGRP,NFNLSB,NTOTAP,         FRLOA472
     2 ASMCOD)                                                            FRLOA473
C                                                                         FRLOA474
  402 CONTINUE                                                            FRLOA475
C                                                                         FRLOA476
  260 CONTINUE                                                            FRLOA477
C                                                                         FRLOA478
C     PRINT LOADS IF DESIRED                                              FRLOA479
C                                                                         FRLOA480
      IF(KPRCHK.EQ.0)GO TO 270                                            FRLOA481
C                                                                         FRLOA482
      IF(IZROS.EQ.0)CALL PRNT(                                            FRLOA483
     1 PS,NINTLD,NFREQ,NINTLD,2,16H SYM  INTGD LOADS   ,4)                FRLOA484
      IF(NASYM.NE.0.AND.IZROA.EQ.0)CALL PRNT(                             FRLOA485
     1 PA,NINTLD,NFREQ,NINTLD,2,16HASYM  INTGD LOADS   ,4)                FRLOA486
C                                                                         FRLOA487
  270 CONTINUE                                                            FRLOA488
C                                                                         FRLOA489
C                                                                         FRLOA490
      IF(KPRCXI.EQ.0)GO TO 280                                            FRLOA491
      IF(IZROS.EQ.1)GO TO 290                                             FRLOA492
      CALL HEADNG                                                         FRLOA493
      WRITE(NOUT,2)IG                                                     FRLOA494
    2 FORMAT(1H ,20X, SYM. UNIT GUST LOADS FOR ORIENTATION NO. ,I4)       FRLOA495
      CALL INTLPR(PS,NINTLD,NFREQ,OMEGA)                                  FRLOA496
  290 CONTINUE                                                            FRLOA497
      IF(NASYM.EQ.0)GO TO 280                                             FRLOA498
      IF(IZROA.EQ.1)GO TO 280                                             FRLOA499
      CALL HEADNG                                                         FRLOA500
      WRITE(NOUT,3)IG                                                     FRLOA501
    3 FORMAT(1H ,20X, ASYM. UNIT GUST LOADS FOR ORIENTATION NO. ,I4)      FRLOA502
      CALL INTLPR(PA,NINTLD,NFREQ,OMEGA)                                  FRLOA503
C                                                                         FRLOA504
  280 CONTINUE                                                            FRLOA505
C                                                                         FRLOA506
C     NOW PLOT SELECTED INTGD LOAD                                        FRLOA507
C                                                                         FRLOA508
      IF(IPLL.EQ.0)GO TO 299                                              FRLOA509
      IF(IPLL.GT.NINTLD)GO TO 299                                         FRLOA510
      IF(IZROS.EQ.1)GO TO 298                                             FRLOA511
C                                                                         FRLOA512
      DO 295 I=1,NFREQ                                                    FRLOA513
      PLOTF(I)=OMEGA(I)/TWOPI                                             FRLOA514
  295 PLOTV(I)=SQRT(PS(1,IPLL,I)**2+PS(2,IPLL,I)**2)                      FRLOA515
      CALL PLT1(NFREQ,PLOTF,PLOTV,1)                                      FRLOA516
      WRITE(NOUT,6)IG,IPLL                                                FRLOA517
    6 FORMAT(1H0,20X,  SYM UNIT GUST LOADS FOR ORIENT ,I4,  LOAD ,I4)     FRLOA518
  298 CONTINUE                                                            FRLOA519
      IF(NASYM.EQ.0)GO TO 299                                             FRLOA520
      IF(IZROA.EQ.1)GO TO 299                                             FRLOA521
```

```
      DO 296 I=1,NFREQ                                                  FRLOA522
  296 PLOTV(I)=SQRT(PA(1,IPLL,I)**2+PA(2,IPLL,I)**2)                    FRLOA523
      CALL PLT1(NFREQ,PLOTF,PLOTV,1)                                    FRLOA524
      WRITE(NOUT,7)IG,IPLL                                              FRLOA525
    7 FORMAT(1H0,20X, ASYM UNIT GUST LOADS FOR ORIENT ,I4,  LOAD ,I4)   FRLOA526
C                                                                       FRLOA527
  299 CONTINUE                                                          FRLOA528
C                                                                       FRLOA529
      DO 293 J=1,NFREQ                                                  FRLOA530
      IF(OMEGA(J).GT.0.062832)GO TO 294                                 FRLOA531
      DO 292 I=1,NINTLD                                                 FRLOA532
      PS(1,I,J)=0.0                                                     FRLOA533
      PS(2,I,J)=0.0                                                     FRLOA534
      PA(1,I,J)=0.0                                                     FRLOA535
      PA(2,I,J)=0.0                                                     FRLOA536
  292 CONTINUE                                                          FRLOA537
  293 CONTINUE                                                          FRLOA538
  294 CONTINUE                                                          FRLOA539
C                                                                       FRLOA540
C     NOW SAVE PS AND PA ON LOAD DATASET                                FRLOA541
C                                                                       FRLOA542
      CALL WRLOAD (8,IG,PS,NOUT,NER)                                    FRLOA543
      CALL WRLOAD (9,IG,PA,NOUT,NER)                                    FRLOA544
C                                                                       FRLOA545
  300 CONTINUE                                                          FRLOA546
C                                                                       FRLOA547
      KPRCXI=ITEMPI                                                     FRLOA548
      KPRCHK=ITEMPK                                                     FRLOA549
C                                                                       FRLOA550
      CALL HEADNG                                                       FRLOA551
      WRITE(NOUT,1)NG                                                   FRLOA552
    1 FORMAT(1H0,20X, UNIT GUST LOADS COMPLETE FOR ,I3,  ORIENTATIONS ) FRLOA553
      IF(NTFS.NE.0.OR.NTFA.NE.0)WRITE(NOUT,11)                          FRLOA554
   11 FORMAT(1H0,20X, ACS ON FOR THIS ANALYSIS )                        FRLOA555
C                                                                       FRLOA556
      RETURN                                                            FRLOA557
      END                                                               FRLOA558
```

```
      SUBROUTINE INTLPR(F,NP,NF,OMEGA)                                  INTLPR 2
      COMMON/777/HEDR(48),NIN,NOUT,ICM,IC,IPRNT,NER                     INTLPR 3
      DIMENSION F(2,NP,NF)                                              INTLPR 4
      DIMENSION FMOD(8)                                                 INTLPR 5
      DIMENSION OMEGA(NF)                                               INTLPR 6
      DATA TWOPI/6.283185/                                              INTLPR 7
    7 FORMAT(1H0,*FREQ HZ*,8(4X,I4,7X))                                 INTLPR 8
    9 FORMAT(1H0,30X,*INT. LOADS - MODULUS*/1H0,30X,*INT. LOAD NO.*)    INTLPR 9
    8 FORMAT(1H ,F7.3,8F15.7)                                           INTLPR10
      IS=1                                                              INTLPR11
      IF=8                                                              INTLPR12
  400 IF(IF.GT.NP)IF=NP                                                 INTLPR13
      IF(IC.LT.ICM)GO TO 401                                            INTLPR14
      CALL HEADNG                                                       INTLPR15
  401 CONTINUE                                                          INTLPR16
      WRITE(NOUT,9)                                                     INTLPR17
      JF=IF-IS+1                                                        INTLPR18
      WRITE(NOUT,7)(J,J=IS,IF)                                          INTLPR19
      IC=IC+6                                                           INTLPR20
      DO 500 I=1,NF                                                     INTLPR21
      FREQ=OMEGA(I)/TWOPI                                               INTLPR22
      IC=IC+1                                                           INTLPR23
      IF(IC.LE.ICM)GO TO 692                                            INTLPR24
      CALL HEADNG                                                       INTLPR25
      WRITE(NOUT,9)                                                     INTLPR26
      WRITE(NOUT,7)(J,J=IS,IF)                                          INTLPR27
      IC=IC+6                                                           INTLPR28
  692 CONTINUE                                                          INTLPR29
      DO 490 J=1,JF                                                     INTLPR30
      JJ=J+IS-1                                                         INTLPR31
  490 FMOD(J)=SQRT(F(1,JJ,I)**2+F(2,JJ,I)**2)                           INTLPR32
  500 WRITE(NOUT,8)FREQ,(FMOD(J),J=1,JF)                                INTLPR33
      IF(IF.GE.NP)GO TO 510                                             INTLPR34
      IS=IF+1                                                           INTLPR35
      IF=IF+8                                                           INTLPR36
      GO TO 400                                                         INTLPR37
  510 CONTINUE                                                          INTLPR38
      RETURN                                                            INTLPR39
      END                                                               INTLPR40
```

```
      SUBROUTINE AERO2 (NOUT,INDSYM,DCP,FZ,FY                          AERO2  2
     1 , DELA,DELX,XIC,CG,CS,FE,SG,YS,ZS,XIJ                            AERO2  3
     2 , ISSTR,COORD,NBARAY,NCARAY,NSBEA,YB,ZB,XIS1,XIS2                AERO2  4
     3 , CN, CM, SPLD, CPR, CPI, CZB, CYB, CNB, CMB )                   AERO2  5
      DIMENSION DELA(1)                                                 AERO2  6
      DIMENSION ISSTR(1),NSBEA(1),NBARAY(1),NCARAY(1),YB(1),ZB(1)       AERO2  7
      DIMENSION XIS1(1),XIS2(1),CG(1),CS(1),FE(1),SG(1),YS(1),ZS(1)     AERO2  8
      DIMENSION XIC(1),XIJ(1),DELX(1),COORD(1)                          AERO2  9
C  ***   COMPUTES  AERODYNAMIC PARAMETERS FOR ALL  LIFTING SURFACES     AERO2 10
C        AND ALL SLENDER BODIES                                         AERO2 11
      COMMON /ARGCOM/ NTI, MODES                                        AROCOM 2
     X , NP, MSTRIP, NSMAX, NCMAX, NTOTAL, NB, MSBE, MBE                AROCOM 3
     Y , ND, NF, NBY, NBZ, NTO, NTP, NTY, NTZ                           AROCOM 4
     1 , NTYS, NTZS, MAXGR, MAXSTR, NSBETO, NSTRIP, KR, XM, REFA, REFC  AROCOM 5
     2 , REFS, FMACH, LINES                                             AROCOM 6
      COMPLEX DCP(1), FZ(1), FY(1), CN(1), CM(1), SPLD(1)               AERO2 13
      COMPLEX CZB(1), CYB(1), CNB(1), CMB(1)                            AERO2 14
      DIMENSION CPR(1), CPI(1)                                          AERO2 15
      COMPLEX CZT, CYT, CMT, CNT, CXT, CLT, FZLB, FYLB                  AERO2 16
      COMPLEX    CZTS, CYTS, CMTS, CNTS, CLTS                           AERO2 17
      REAL       KR                                                     AERO2 18
   10 FORMAT  ( 1H1 /// 10X, 21H***    PRESSURES    *** //              AERO2 19
     1           13X, 4HKR = , F8.4 / )                                 AERO2 20
   20 FORMAT  ( //4X,8HPANEL    , 10H  STRIP     ,                      AERO2 21
     1          10H    BOX   ,7X,3HXOC,8X,1HX,11X,1HY,11X,1HZ,15X,      AERO2 22
     2          10H PRESSURES/ )                                        AERO2 23
   30 FORMAT  ( 1H1 /// 10X, 21HDEFINITION OF SYMBOLS    //   10X,      AERO2 24
     1          18HS       =   SEMI-SPAN/10X, 25HC-BAR =  REFERENCE CHORD /) AERO2 25
   40 FORMAT  ( 10X, 20HL       =  BODY LENGTH / 10X, 45HF-Z/Q =  (VERTICAAERO2 26
     1L FORCE)/(DYNAMIC PRESSURE)              / 10X, 45HF-Y/Q =  (LATERALAERO2 27
     2  FORCE)/(DYNAMIC PRESSURE)    // )                               AERO2 28
   50 FORMAT (1H1// 7H  STRIP, 5X, 1HY,  9X, 1HZ,                       AERO2 29
     1     5X,3HY/S,10X,16HLIFT COEFFICIENT, 7X,18HMOMENT COEFFICIENT,  AERO2 30
     2     6X, 18HCENTER OF PRESSURE / 64X, 18H1/4 CHORD OF STRIP  /    AERO2 31
     3      41X, 18HREAL          IMAG., 5X, 18HREAL          IMAG. ,   AERO2 32
     4       9X, 14HREAL      IMAG.   / )                               AERO2 33
   60 FORMAT ( 1H1/// 6H  BODY, 6X,1HY,11X,1HZ,5X,                      AERO2 34
     1  3HX/L,4X,49HRUNNING LOAD (VERTICAL)   RUNNING LOAD (LATERAL)  / AERO2 35
     2  7H  ELEM., 36X, 38H(DF-Z/Q) / DX           (DF-Y/Q) / DX      / AERO2 36
     3          39X, 45HREAL          IMAG.         REAL          IMAG./) AERO2 37
   70 FORMAT  (  6X, I4, 6X,I4,6X,I4,3X, 4F12.5,2F12.6 )                AERO2 38
   72 FORMAT  ( I5, 3F10.4, 4F12.6, 2F10.4)                             AERO2 39
   80 FORMAT  ( // 10X, 15HSIGN CONVENTION  //  10X,                    AERO2 40
     1          39HTHE FOLLOWING TWO SIGN CONVENTIONS HOLD  //  10X,    AERO2 41
     2          76H(1)   FORCES  AND DEFLECTIONS POSITIVE DOWN ON HORIZONTALAERO2 42
     3 SURFACES OR Z-BODIES  / 15X, 54HAND POSITIVE OUTBOARD ON VERTICALAERO2 43
     4 SURFACES OR Y-BODIES  //10X,                                     AERO2 44
     5          76H(2)  FORCES  AND DEFLECTIONS POSITIVE  UP  ON HORIZONTALAERO2 45
     6 SURFACES OR Z-BODIES  / 15X, 54HAND POSITIVE INBOARD  ON VERTICALAERO2 46
     7 SURFACES OR Y-BODIES  //    )                                    AERO2 47
   90 FORMAT  (/// 25X, 14HTOTALS ON BODY , I3  // 10X, 7HF-Z/Q = ,     AERO2 48
     1          2F15.6, 10X, 7HM-Z/Q = , 2F15.6    / 10X, 7HF-Y/Q = ,   AERO2 49
```

```
     2          2F15.6, 10X, 7HM-Y/Q = , 2F15.6      / )
 100 FORMAT ( ///25X,26HTOTALS ON LIFTING SURFACES )
 110 FORMAT  ( /10X, 7HCZ     =, 2F15.6, 10X, 7HCY     =, 2F15.6 / 10X,
     1           7HCM     =, 2F15.6, 10X, 7HCN     =, 2F15.6 / 10X,
     2           7HCSL    =, 2F15.6 / )
 120 FORMAT  (// 25X, 27HTOTALS ON ENTIRE  AIRCRAFT       )
 130 FORMAT  ( I5, F12.4, 3X, 4E14.5 )
 140 FORMAT  ( 1H1 )
 150 FORMAT  ( // 48H  STATION  SPANWISE    SPAN LOAD  (C*CL)/(C-BAR)
     1           /  46H            COORD.         REAL            IMAG. /)
      WRITE  (NOUT,10)  KR
C
      WRITE (NOUT,20)
      I2         =   0
      LP         =   1
      LPAGE      =   1
      DO  170   J = 1, NSTRIP
      I1         =   I2 + 1
      I2         =   I2 + NCARAY(LP)
      DO  160   I = I1, I2
      XCC   = (XIC(I)-XIJ(J)) / CS(J)
      IF (YS(J).EQ.0.0) DCP(I) = DCP(I)*2.0
      DCP(I) = DCP(I)/DELA(I)
      WRITE  (NOUT,70)  LP, J, I, XCC, XIC(I), YS(J), ZS(J), DCP(I)
      IF   (LPAGE*LINES .NE. I)  GO TO  160
      LPAGE      =   LPAGE + 1
      WRITE  (NOUT,20)
 160  CONTINUE
      IF  (I2.EQ.NBARAY(LP))   LP=LP+1
 170  CONTINUE
C
C   ***
C
      DELT = 1.0
      IF  (INDSYM.EQ.0)  DELT=-1.0
      SYMA = 1.0+DELT
      SYMB = 1.0-DELT
      DO 190 I=1,NSTRIP
      CPR(I) = 0.0
      CPI(I) = 0.0
 190  CONTINUE
      WRITE   (NOUT,30)
      IF  (NB.NE.0)  WRITE(NOUT,40)
      WRITE   (NOUT,80)
      I2    = 0
      LP    = 1
      WRITE  (NOUT,50)
      DO   260   J=1,NSTRIP
      I1    = I2+1
      I2    = I2+NCARAY(LP)
      IF  (I2.EQ.NBARAY(LP))   LP=LP+1
      XI14S = 0.25*CS(J) + XIJ(J)
```

```
      CN(J)= (0.0,0.0)                                                  AER02102
      CM(J)= (0.0,0.0)                                                  AER02103
      SPLD(J)   =   0.0                                                 AER02104
      DO  220  I=I1,I2                                                  AER02105
      CN(J)= CN(J) + DCP(I)*DELX(I)                                     AER02106
      CM(J)= CM(J) + DCP(I)*DELX(I)*(XIC(I) - XI14S)                    AER02107
  220 CONTINUE                                                          AER02108
      YOS   = YS(J)/REFS                                                AER02109
      CH2      = CS(J)**2                                               AER02110
      JL         =    ISSTR(J)                                          AER02111
      SPLD(JL) = SPLD(JL) + CN(J) / REFC                                AER02112
      CN(J)   = CN(J)/CS(J)                                             AER02113
      CM(J)   =-CM(J)/CH2                                               AER02114
      ABSCNJ = SQRT((REAL(CN(J)))**2 + (AIMAG(CN(J)))**2)               AER02115
      IF    ( ABSCNJ .LE.0.0000001)    GO TO  230                       AER02116
      CPR(J) =-REAL(CM(J))/REAL(CN(J)) + 0.25                           AER02117
      IF  (KR.LE.0.0001)   GO TO  240                                   AER02118
      CPI(J) =-AIMAG(CM(J))/AIMAG(CN(J))+0.25                           AER02119
      GO TO  250                                                        AER02120
  230 CONTINUE                                                          AER02121
      CPR(J) = 0.0                                                      AER02122
  240 CONTINUE                                                          AER02123
      CPI(J) = 0.0                                                      AER02124
  250 CONTINUE                                                          AER02125
      WRITE   (NOUT, 72)   J,YS(J), ZS(J), YOS,CN(J),CM(J),CPR(J),CPI(J) AER02126
  260 CONTINUE                                                          AER02127
      WRITE    (NOUT,140)                                               AER02128
      WRITE    (NOUT,150)                                               AER02129
      DO   270   JL = 1, MAXSTR                                         AER02130
      WRITE   (NOUT,130)   JL, COORD(JL), SPLD(JL)                      AER02131
  270 CONTINUE                                                          AER02132
      IF  (NB.EQ.0) GO TO 300                                           AER02133
      L2    = 0                                                         AER02134
      DO   290   N=1, NB                                                AER02135
      CZB(N)  =  (0.0,0.0)                                              AER02136
      CYB(N)  =  (0.0,0.0)                                              AER02137
      CMB(N)  =  (0.0,0.0)                                              AER02138
      CNB(N)  =  (0.0,0.0)                                              AER02139
      L1    = L2+1                                                      AER02140
      L2    = L2 + NSBEA(N)                                             AER02141
      SBL       =   XIS2(L2) - XIS1(L1)                                 AER02142
      WRITE  (NOUT,60)                                                  AER02143
      DO 280 LB=L1,L2                                                   AER02144
      IF  (YB(LB).EQ.0.0) FZ(LB) = FZ(LB)*2.0                           AER02145
      IF  (YB(LB).EQ.0.0) FY(LB) = FY(LB)*2.0                           AER02146
      LX        = LX+1                                                  AER02147
      XSB   = 0.5*(XIS1(LB) + XIS2(LB))                                 AER02148
      DXSB  = XIS2(LB) - XIS1(LB)                                       AER02149
      XOL       =  (XSB -XIS1(L1))/ SBL                                 AER02150
      FZLB  = FZ(LB)/ DXSB                                              AER02151
      FYLB  = FY(LB)/ DXSB                                              AER02152
      WRITE   (NOUT, 72)   LB, YB(N), ZB(N), XOL, FZLB, FYLB            AER02153
```

```
      CZB(N) = CZB(N) + FZ(LB)                                          AERO2154
      CYB(N) = CYB(N) + FY(LB)                                          AERO2155
      CMB(N) = CMB(N) - FZ(LB)*(XSB-XIS1(L1))                           AERO2156
      CNB(N) = CNB(N) - FY(LB)*(XSB-XIS1(L1))                           AERO2157
  280 CONTINUE                                                          AERO2158
      CZB(N) = CZB(N)/REFA                                              AERO2159
      CYB(N) = CYB(N)/REFA                                              AERO2160
      CMB(N) = CMB(N)/(REFA*REFC)                                       AERO2161
      CNB(N) = CNB(N)/(REFA*REFC)                                       AERO2162
      WRITE  (NOUT, 90)   N, CZB(N), CMB(N), CYB(N), CNB(N)             AERO2163
  290 CONTINUE                                                          AERO2164
  300 CONTINUE                                                          AERO2165
      WRITE  (NOUT,140)                                                 AERO2166
      CZT  = (0.0,0.0)                                                  AERO2167
      CYT  = (0.0,0.0)                                                  AERO2168
      CMT  = (0.0,0.0)                                                  AERO2169
      CNT  = (0.0,0.0)                                                  AERO2170
      CLT  = (0.0,0.0)                                                  AERO2171
      DO  320  J=1,NSTRIP                                               AERO2172
      CH2  = CS(J)**2                                                   AERO2173
      XI14S= 0.25*CS(J) + XIJ(J)                                        AERO2174
      CMULT= 2.0*EE(J)*CG(J)                                            AERO2175
      SMULT= 2.0*EE(J)*SG(J)                                            AERO2176
      GUCJ = 1.0                                                        AERO2177
      IF   (ABS(YS(J)).LE.0.0001 . AND .ABS(CG (J)).LE.0.0001)  GUCJ=0.5 AERO2178
      CXT  = CH2*CM(J)  - CS(J)*CN(J)*(XI14S-XM)                        AERO2179
      CZT  = CZT + CS(J)*CN(J)*CMULT  * GUCJ                            AERO2180
      CYT  = CYT - CS(J)*CN(J)*SMULT  * GUCJ                            AERO2181
      CMT  = CMT + CXT  * CMULT         * GUCJ                          AERO2182
      CNT  = CNT - CXT  * SMULT         * GUCJ                          AERO2183
      CLT  = CLT+CS(J)*CN(J)*(YS(J)*CG (J)+ZS(J)*SG (J))*2.0*EE(J)*GUCJ AERO2184
  320 CONTINUE                                                          AERO2185
      WRITE  (NOUT,100)                                                 AERO2186
      CZT  = CZT/REFA                                                   AERO2187
      CZTS = SYMA * CZT                                                 AERO2188
      CYT  = CYT/REFA                                                   AERO2189
      CYTS = SYMB * CYT                                                 AERO2190
      CMT  = CMT/(REFA*REFC)                                            AERO2191
      CMTS = SYMA * CMT                                                 AERO2192
      CNT  = CNT/(REFA*REFC)                                            AERO2193
      CNTS = SYMB * CNT                                                 AERO2194
      CLTS = SYMB * CLT  /  (2.0*REFA*REFS)                             AERO2195
      WRITE  (NOUT,110)   CZTS, CYTS, CMTS, CNTS, CLTS                  AERO2196
      IF  (NB.EQ.0) GO TO 380                                           AERO2197
      L1    = 0                                                         AERO2198
      L2    = 0                                                         AERO2199
      J1    = 1                                                         AERO2200
      J2    = NBZ                                                       AERO2201
      DO   330   J=J1,J2                                                AERO2202
      GLCB = 1.0                                                        AERO2203
      IF   (ABS(YB( J)).LE.0.0001)   GLCB = 0.5                         AERO2204
      L1    = L2+1                                                      AERO2205
```

```
      L2    = L2+NSBEA(J)                                               AER02206
      CZT   = CZT + CZB(J) * GLCB                                       AER02207
      CMT   = CMT + (CMB(J)-CZB(J)*(XIS1(L1)-XM)/REFC) * GLCB           AER02208
      CLT   = CLT + REFA * CZB(J) * YB(J) * GLCB                        AER02209
  330 CONTINUE                                                          AER02210
      L1    = 0                                                         AER02211
      L2    = 0                                                         AER02212
      J1    = NB-NBY+1                                                  AER02213
      J2    = NB                                                        AER02214
      J1M1  = J1-1                                                      AER02215
      IF   (J1M1.EQ.0)  GO TO  360                                      AER02216
      DO  350  JX=1,J1M1                                                AER02217
      L2    = L2+NSBEA(JX)                                              AER02218
  350 CONTINUE                                                          AER02219
  360 CONTINUE                                                          AER02220
      DO  370  J=J1,J2                                                  AER02221
      GLCB  = 1.0                                                       AER02222
      IF   (ABS(YB( J)).LE.0.0001)  GLCB = 0.5                          AER02223
      L1    = L2+1                                                      AER02224
      L2    = L2+NSBEA(J)                                               AER02225
      CYT   = CYT + CYB(J) * GLCB                                       AER02226
      CNT   = CNT + (CNB(J)-CYB(J)*(XIS1(L1)-XM)/REFC) * GLCB           AER02227
      CLT   = CLT - REFA * CYB(J) * ZB(J) * GLCB                        AER02228
  370 CONTINUE                                                          AER02229
  380 CONTINUE                                                          AER02230
      CZT   = SYMA * CZT                                                AER02231
      CYT   = SYMB * CYT                                                AER02232
      CMT   = SYMA * CMT                                                AER02233
      CNT   = SYMB * CNT                                                AER02234
      CLT   = SYMB * CLT / (2.0*REFS*REFA)                              AER02235
      WRITE  (NOUT,120)                                                 AER02236
      WRITE  (NOUT,110)  CZT , CYT , CMT , CNT , CLT                    AER02237
      RETURN                                                            AER02238
      END                                                               AER02239
```

```
      SUBROUTINE CTRIM(AMODNO,EMBAR,CAY,PS,PA,THRUST)
C
      DIMENSION THRUST(1)
      DIMENSION AMODNO(20), EMBAR(1), CAY(1)
      COMMON NAA,A(1)
      COMMON /XTF/NF(100)
      EQUIVALENCE (NF(6),NSYM),     (NF(7),NASYM),     (NF(8),NMODES)
     1      ,     (NF(10),NENGS),   (NF(32),NINTLD)
     2      ,     (NF(43),NBOX),    (NF(44),NSBE)
C
      COMMON /ZZZ/HEDR(48),NIN,NOUT,KROW,LINES,IPRNT,NER
C
      DATA CORERQ/8H  TRIM  /
C
      NMS = 2
      IF (AMODNO(4).NE.0) NMS = NMS+AMODNO(5)-AMODNO(4)+1
      NMA = 3
      IF (AMODNO(14).NE.0) NMA = NMA+AMODNO(15)-AMODNO(14)+1
C
      L1  = NAA
      L2  = L1  + NMODES
      L14 = L2 + NMODES
      L15 = L14 + NINTLD*NENGS
      L3 = L15 + NSYM*NENGS
      L4  = L3  + NMS*NMS
      L5  = L4  + NMS
      LSYM = L5 + 2*NSYM*NSYM
      L6  = L3
      L7  = L6  + NMA*NMA
      L8  = L7  + NMA
      LASM = L8 + 2*NASYM*NASYM
      L11 = L3
      LPIQ = L11 + NINTLD*NMODES
      L12 = L3
      L13 = L12 + 2*NINTLD*NSYM
      LF = L13 + 2*NINTLD*NASYM
C
      LMAX = MAX0(LSYM,LASM,LF,LPIQ)
C
      WRITE (NOUT,999)LMAX,CORERQ
  999 FORMAT(1H0,20X,I10,  WORDS OF CORE RQD FOR STEP +++ ,A10, +++ )
C
      CALL TRIM (AMODNO,EMBAR,CAY,NSYM,NASYM,NMODES,NINTLD,NMS,NMA
     1 , A(L1),A(L2),A(L3),A(L4),A(L5),A(L6),A(L7),A(L8)
     2 , PS,PA,A(L11),A(L12),A(L13),A(L14),A(L15)
     3 , THRUST,IVBW)
C
      L2  = L1  + NMODES
      L3  = L2  + 2*NBOX
      L4  = L3  + 2*NSBE
      L5  = L4  + 2*NSBE
      L6  = L5  + 2*MAX0(NBOX,NSBE)*MAX0(NSYM,NASYM)
```

```
C
      CALL SPANLD (NOUT,NER,IVBW,NBCX,NSBE
     1 , A(L1),A(L2),A(L3),A(L4),A(L5),A(L5),A(L6) )
C
      RETURN
      END
```

```
      SUBROUTINE SPANLD (NOUT,NER,IVBW,NBOX,NSBE,Q,DCP,FZ,FY,D,DB,A)      SPANLD 2
C                                                                         SPANLD 3
      DIMENSION Q(1), A(1)                                                SPANLD 4
      COMPLEX DCP(1), FZ(1), FY(1)                                        SPANLD 5
      COMPLEX D(NBOX,1), DB(NSBE,1)                                       SPANLD 6
C                                                                         SPANLD 7
      COMMON /XTE/NF(100)                                                 SPANLD 8
      EQUIVALENCE (NF(6),NSYM),      (NF(7),NASYM)                        SPANLD 9
      EQUIVALENCE (NF(57),INDSYM)                                         SPANLD10
C                                                                         SPANLD11
      COMMON /ARCCOM/ NTI, MODES                                          AROCOM 2
     X , NP, MSTRIP, NSMAX, NCMAX, NTOTAL, NB, MSBE, MBE                  AROCOM 3
     Y , ND, NE, NBY, NBZ, NTO, NTP, NTY, NTZ                             AROCOM 4
     1 , NTYS, NTZS, MAXGR, MAXSTR, NSBETC, NSTRIP, KP, XM, REFA, REFC    AROCOM 5
     2 , REFS, FMACH, LINES                                               AROCOM 6
C                                                                         SPANLD13
      DIMENSION IHD(50),RHD(50)                                           SPANLD14
      EQUIVALENCE (IHD(1),RHD(1))                                         SPANLD15
C                                                                         SPANLD16
      CALL RDAERO (4HHEAD,0,IHD,NOUT,NER)                                 SPANLD17
C                                                                         SPANLD18
      NP = IHD(6)                                                         SPANLD19
      NBY = NB                                                            SPANLD20
      NBZ = NB                                                            SPANLD21
      NSTRIP = IHD(9)                                                     SPANLD22
      MAXSTR = IHD(10)                                                    SPANLD23
      NP     = IHD(11)                                                    SPANLD24
      FMACH  = RHD(21)                                                    SPANLD25
      REFA   = RHD(22)                                                    SPANLD26
      REFS   = RHD(23)                                                    SPANLD27
      REFC   = RHD(24)                                                    SPANLD28
      XM     = RHD(25)                                                    SPANLD29
C                                                                         SPANLD30
      CALL RDAERO (4HGEOM,0,A,NOUT,NER)                                   SPANLD31
C                                                                         SPANLD32
      L1  = 1                                                             SPANLD33
      L2  = L1  + NBOX                                                    SPANLD34
      L3  = L2  + NBOX                                                    SPANLD35
      L4  = L3  + NBOX                                                    SPANLD36
      L5  = L4  + NSTRIP                                                  SPANLD37
      L6  = L5  + NSTRIP                                                  SPANLD38
      L7  = L6  + NSTRIP                                                  SPANLD39
      L8  = L7  + NSTRIP                                                  SPANLD40
      L9  = L8  + NSTRIP                                                  SPANLD41
      L10 = L9  + NSTRIP                                                  SPANLD42
      L11 = L10 + NSTRIP                                                  SPANLD43
      L12 = L11 + NSTRIP                                                  SPANLD44
      L13 = L12 + MAXSTR                                                  SPANLD45
      L14 = L13 + NP                                                      SPANLD46
      L15 = L14 + NP                                                      SPANLD47
      L16 = L15 + NB                                                      SPANLD48
      L17 = L16 + NP                                                      SPANLD49
```

```
      L18 = L17 + NR                                                    SPANLD50
      CALL  HEADNG                                                      SPANLD51
C                                                                       SPANLD52
      CALL RDAERO (4HGEOS,0,A(L18),NOUT,NER)                            SPANLD53
C                                                                       SPANLD54
      L19 = L18 + NSBE                                                  SPANLD55
      N = L19-1                                                         SPANLD56
      M = N+2*NSBE                                                      SPANLD57
      DO 50 I=1,NSBE                                                    SPANLD58
      N = N+1                                                           SPANLD59
      M = M+1                                                           SPANLD60
   50 A(N) = A(M)                                                       SPANLD61
C                                                                       SPANLD62
      CALL RDAERO (4HDPSP,IVBW,D,NOUT,NER)                              SPANLD63
C                                                                       SPANLD64
      DO 100 I=1,NBOX                                                   SPANLD65
      DCP(I) = 0.0                                                      SPANLD66
      DO 100 J=1,NSYM                                                   SPANLD67
  100 DCP(I) = DCP(I) +      D(I,J)*Q(J)                                SPANLD68
C                                                                       SPANLD69
      IF (NSBE.EQ.0) GO TO 150                                          SPANLD70
C                                                                       SPANLD71
      CALL RDAERO (4HDZSP,IVBW,DB,NOUT,NER)                             SPANLD72
C                                                                       SPANLD73
      DO 110 I=1,NSBE                                                   SPANLD74
      FZ(I) = 0.0                                                       SPANLD75
      DO 110 J=1,NSYM                                                   SPANLD76
  110 FZ(I) = FZ(I) +      DB(I,J)*Q(J)                                 SPANLD77
C                                                                       SPANLD78
      CALL RDAERO (4HDYSP,IVBW,DB,NOUT,NER)                             SPANLD79
C                                                                       SPANLD80
      DO 120 I=1,NSBE                                                   SPANLD81
      FY(I) = 0.0                                                       SPANLD82
      DO 120 J=1,NSYM                                                   SPANLD83
  120 FY(I) = FY(I) +      DB(I,J)*Q(J)                                 SPANLD84
C                                                                       SPANLD85
  150 L20 = L19 + NSBE                                                  SPANLD86
      L21 = L20 + 2*NSTRIP                                              SPANLD87
      L22 = L21 + 2*NSTRIP                                              SPANLD88
      L23 = L22 + 2*NSTRIP                                              SPANLD89
      L24 = L23 + 2*NSTRIP                                              SPANLD90
      L25 = L24 + 2*NSTRIP                                              SPANLD91
      L26 = L25 + 2*NSTRIP                                              SPANLD92
      L27 = L26 + 2*NB                                                  SPANLD93
      L28 = L27 + 2*NB                                                  SPANLD94
C                                                                       SPANLD95
      CALL AERO2 (NOUT,INDSYM,DCP,FZ,FY                                 SPANLD96
     1 , A(L1),A(L2),A(L3),A(L4),A(L5),A(L6),A(L7),A(L8),A(L9),A(L10)   SPANLD97
     2 , A(L11),A(L12),A(L13),A(L14),A(L15),A(L16),A(L17),A(L18),A(L19) SPANLD98
     3 , A(L20),A(L21),A(L22),A(L23),A(L24),A(L25),A(L26),A(L27),A(L28))SPANLD99
C                                                                       SPANL100
      IF (INDSYM.EQ.1) GO TO 200                                        SPANL101
```

```
      IF (NASYM.EQ.0) GO TO 200
      CALL RDAERO (4HDPAP,IVBW,D,NOUT,NER)
      DO 160 I=1,NBOX
      DCP(I) = 0.0
      DO 160 J=1,NASYM
      JJ = J+NSYM
  160 DCP(I) = DCP(I) +        D(I,J)*Q(JJ)
C
      IF (NSBE.EQ.0) GO TO 200
C
      CALL RDAERO (4HDZAP,IVBW,DB,NOUT,NER)
C
      DO 170 I=1,NSBE
      FZ(I) = 0.0
      DO 170 J=1,NASYM
      JJ = J+NSYM
  170 FZ(I) = FZ(I) +      DB(I,J)*Q(JJ)
C
      CALL RDAERO (4HDYAP,IVBW,DB,NOUT,NER)
C
      DO 180 I=1,NSBE
      FY(I) = 0.0
      DO 180 J=1,NASYM
      JJ = J+NSYM
  180 FY(I) = FY(I) +      DB(I,J)*Q(JJ)
C
      CALL AERO2 (NOUT,INOSYM,DCP,FZ,FY
     1 , A(L1),A(L2),A(L3),A(L4),A(L5),A(L6),A(L7),A(L8),A(L9),A(L10)
     2 , A(L11),A(L12),A(L13),A(L14),A(L15),A(L16),A(L17),A(L18),A(L19)
     3 , A(L20),A(L21),A(L22),A(L23),A(L24),A(L25),A(L26),A(L27),A(L28))
C
  200 RETURN
      END
```

```
      SUBROUTINE TRIM(AMODNO,EMBAR,CAY,NSYM,NASYM,NMODES,NINTLD,NMS,NMA  TRIM    2
     1 , Q,QDOT,ZSYM,FSYM,DSYM,ZASYM,FASYM,DASYM                         TRIM    3
     2 , PS,PA,PIQ,PAQS,PAQA,THRLOD,THRGNF,THRUST                        TRIM    4
     3 , IVBW )                                                          TRIM    5
C                                                                        TRIM    6
      COMMON/ZZZ/CASE(48),NIN,NOUT,KROW,LINES,IPRNT,NFR                  TRIM    7
      COMMON/XTF/NFLAGS(100)                                             TRIM    8
C                                                                        TRIM    9
      EQUIVALENCE (NFLAGS(10),NENGS)                                     TRIM   10
      EQUIVALENCE (NFLAGS(12),BZERO), (NFLAGS(13),RHO0)                  TRIM   11
     1 , (NFLAGS(14),SIGMA), (NFLAGS(16),IUNITS), (NFLAGS(17),VKFAS)     TRIM   12
     1 , (NFLAGS(19),ALT)                                                TRIM   13
     2 , (NFLAGS(18),VTFPS)                                              TRIM   14
     2 , (NFLAGS(23),KPRCHK)                                             TRIM   15
     4 , (NFLAGS(41),NK),      (NFLAGS(42),NG)                           TRIM   16
      EQUIVALENCE (NFLAGS(51),AN),   (NFLAGS(52),ZDOT),(NFLAGS(53),RTURN)TRIM   17
     1 , (NFLAGS(54),KMAN),(NFLAGS(55),AB),(NFLAGS(56),AC)               TRIM   18
     1 , (NFLAGS(57),INDSYM)                                             TRIM   19
     3 , (NFLAGS(100),SIZFCT)                                            TRIM   20
     4 , (NFLAGS(99),RBRADF)                                             TRIM   21
C                                                                        TRIM   22
      DIMENSION ZSYM(NMS,NMS)                                            TRIM   23
      DIMENSION ZASYM(NMA,NMA)                                           TRIM   24
      DIMENSION VFLUN(3)                                                 TRIM   25
      DIMENSION EMBAR(NMODES,NMODES)                                     TRIM   26
      DIMENSION CAY(1)                                                   TRIM   27
      DIMENSION FSYM(NMS)                                                TRIM   28
      DIMENSION FASYM(NMA)                                               TRIM   29
      DIMENSION Q(NMODES)                                                TRIM   30
      DIMENSION QDOT(NMODES)                                             TRIM   31
      DIMENSION DSYM(2,NSYM,NSYM)                                        TRIM   32
      DIMENSION DASYM(2,NASYM,NASYM)                                     TRIM   33
      DIMENSION AMODNO(20)                                               TRIM   34
      DIMENSION PS(NINTLD)                                               TRIM   35
      DIMENSION PA(NINTLD)                                               TRIM   36
      DIMENSION PIQ(NINTLD,NMODES)                                       TRIM   37
      DIMENSION PAQS(2,NINTLD,NSYM)                                      TRIM   38
      DIMENSION PAQA(2,NINTLD,NASYM)                                     TRIM   39
      DIMENSION THRLOD(NINTLD,1), THRGNF(NSYM,1)                         TRIM   40
      DIMENSION THRUST(1)                                                TRIM   41
C                                                                        TRIM   42
      COMMON/DISK2/ND2,ITBL2(843),NRECSA,IBUMP,NKD,VOBWS(20)             DISK2   2
C                                                                        TRIM   44
   10 FORMAT(1H0,*SINGULAR MATRIX FOR IVBW=*,I4,5X,*VOBW= *,E16.8/1H0,   TRIM   45
     1 4H    **,13(4H****) )                                             TRIM   46
   11 FORMAT(1H0,30X,*TRIMMED SYMMETRIC FLIGHT SOLNS*/1H0,               TRIM   47
     1 10X, R.B. PLUNGE       MODE = ,I4/1H ,                            TRIM   48
     1 10X, R.B. PITCH        MODE = ,I4/1H ,                            TRIM   49
     1 10X, SYMMETRIC TRIM    MODE = ,I4/1H ,                            TRIM   50
     1 10X,*SYM R.B. JIG      MODE =*,I4/1H ,                            TRIM   51
     1 10X,*1ST SYM ELASTIC MODE =*,I4/1H ,                              TRIM   52
     1 10X,*LST SYM ELASTIC MODE =*,I4/1H ,                              TRIM   53
```

```
     1 10X,*TRUE VELOCITY      =*,F10.3,* FPS*/1H ,
     1 10X,*DYNAMIC PRESSURE =*,F10.3,* PSI*/1H ,
     1 10X,*LOAD FACTOR        =*,F10.3/1H ,
     1 10X, PITCH RATE         = ,F10.4,  RAD/SEC )
 111  FORMAT(1H0,30X,*TRIMMED ANTISYMMETRIC FLIGHT SOLNS*/1H0,
     1 10X, R.B. ROLL          MODE = ,I4/1H ,
     1 10X, R.B. LATERAL       MODE = ,I4/1H ,
     1 10X, R.B. YAW           MODE = ,I4/1H ,
     1 10X, ROLL  TRIM         MODE = ,I4/1H ,
     1 10X, YAW   TRIM         MODE = ,I4/1H ,
     1 10X,*ASYM R.B. JIG      MODE =*,I4/1H ,
     1 10X,*1ST ASYM ELASTIC MODE =*,I4/1H ,
     1 10X,*LST ASYM ELASTIC MODE =*,I4/1H ,
     1 10X,*TRUE VELOCITY      =*,F10.3,* FPS*/1H ,
     1 10X,*DYNAMIC PRESSURE =*,F10.3,* PSI*/1H ,
     1 10X, BANK ANGLE         = ,F10.3,  DEG /1H ,
     1 10X,*CLIMB ANGLE        =*,F10.3,* DEG*/1H ,
     1 10X, ROLL RATE          = ,F10.4,  RAD/SEC /1H ,
     1 10X, YAW  RATE          = ,F10.4,  RAD/SEC /1H ,
     1 10X, TURN RADIUS        = ,F12.2, FT. )
  30 FORMAT(1H ,I4,2E16.8)
  40  FORMAT(1H0,1X, GENERALIZED COORDINATES /1H0,12X, Q ,13X, QDOT )
  50  FORMAT(1H0,20X, AEROELASTIC TRIM SOLUTIONS /1H0,
     1 20X, VEL =   ,F8.2,1X,A4,10X,   ALT =   ,F10.2,   FT. )
 60   FORMAT(1H0,30X,*IVBW = *,I3)
  4   FORMAT(1H0,10X, ALPHA TRIM          = ,F10.4,   DEG /1H ,
     1 10X, DELTA PITCH TRIM = ,F10.4,   DEG )
  5   FORMAT(1H0,10X, BETA  TRIM          = ,F10.4,   DEG /1H ,
     1 10X, DELTA YAW   TRIM = ,F10.4,   DEG /1H ,
     1 10X, DELTA ROLL  TRIM = ,F10.4,   DEG )
C
      DATA VELUN/4H FPS,4HKEAS,4H MPH/
      DATA GEE/386.088/
      DATA RAD/0.017453/
C
C
      FACTOR=SIZFCT**2
      IF(RBRADF.EQ.0)RBRADF=1.0
      IUNIT=IABS(IUNITS)
      LINES=15
C
C
      CALL HEADNG
      WRITE(NOUT,50)VKEAS,VELUN(IUNIT),ALT
      INDSYM=1
      IF(KMAN.EQ.2)INDSYM=0
C
C     INDSYM =0 IS AN UNSYMMETRIC TRIM CASE
C     INDSYM =1 IS A SYMMETRIC TRIM CASE
C
      OVERV=1./(VTFPS*12.)
      DYNP=SIGMA*RHOO*VTFPS**2*FACTOR/288.
```

```
      AMULTA=-DYNP
      IF(KMAN.EQ.0)AN=1.0
      QBAR=0.
      PBAR=0.
      RBAR=0.
      AB = 0.0
      AC = 0.0
      RTURN=0.0
C
C
      IF(INDSYM.EQ.1)GO TO 100
      AB=ACOS(1.0/AN)/RAD
      IF(AB.EQ.0)GO TO 1017
      RTURN=(VTFPS**2)*12./(GEE*TAN(AB*RAD))
      AC = ASIN(ZDOT/VTFPS)/RAD
      QBAR=AN*GEE*(1.-1./AN**2)*OVERV
      PBAR=-AC*RAD*GEE*TAN(AB*RAD)*OVERV
      RBAR=GEE*SIN(AB*RAD)*OVERV
      GO TO 101
C
 100  CONTINUE
C
      IF(KMAN.NE.0)QBAR=(AN-1.0)*GEE*OVERV
 101  CONTINUE
      ANGEE=AN*GEE/SIZFCT
C
C     CORRECT ANG. VEL. FOR RB ROT MODES NOT AT 1 RADIAN
C
      QBARP=QBAR
      PBARP=PBAR
      RBARP=RBAR
C
      QBAR=QBAR*RBRADF
      PBAR=PBAR*RBRADF
      RBAR=RBAR*RBRADF
C
C     READ IN LOWST K GENLZD SYM AERO INTO DSYM
C
C     VCBW=1/K    FOR THIS AERO SET
C
      IVBW = 1
      VCBW = VCBWS(1)
      DO 95  I=2,NK
      IF (VCBWS(I).LT.VCBW) GO TO 95
      IVBW = I
      VCBW = VCBWS(I)
  95  CONTINUE
      IVBWN = 1
      IF (IVBWN.EQ.IVBW) IVBWN=2
      VCBWN = VCBWS(IVBWN)
      DO 98 I=1,NK
      IF (I.EQ.IVBW) GO TO 98
```

```
      IF (VOBWS(I).LT.VOBWN) GO TO 98
      IVBWN = I
      VOBWN = VOBWS(I)
   98 CONTINUE
      NWD = 2*NSYM*NSYM
      LOC=1+(IVBW-1)*IBUMP
      CALL READMS(ND2,DSYM ,NWD,LOC)
C
C     FORM EQNS OF EQUILIBRIUM
C
      DO 300 I=1,NMS
      FSYM(I)=0.
      DO 300 J=1,NMS
  300 ZSYM(I,J)=0.
C
C     FORM Z MATRIX
C
      MH=AMODNO(1)
      MA=AMODNO(2)
      MT=AMODNO(6)
      MJ=AMODNO(7)
      MEF=AMODNO(4)
      MEL=AMODNO(5)
      WRITE(NOUT,11) MH,MA,MT,MJ,MEF,MEL,VTFPS,DYNP,AN,QBARP
      LINES=LINES+16
      DELX=EMBAR(MH,MA)/EMBAR(MH,MH)
C
C
      IF(KPRCHK.EQ.0)GO TO 571
      CALL PRNT(DSYM,NSYM,NSYM,NSYM,2,12HDSYMREALAERO   ,3)
      CALL PRNT(EMBAR(1,MH),NSYM,1,NMODES,1,12HEMBAR(MH)    ,3)
  571 CONTINUE
C
      IF(MT.EQ.0)GO TO 1015
C
C
      ZSYM(1,1)=AMULTA*DSYM(1,MH,MA)
      ZSYM(1,2)=AMULTA*DSYM(1,MH,MT)
      ZSYM(2,1)=AMULTA*DSYM(1,MA,MA)
      ZSYM(2,2)=AMULTA*DSYM(1,MA,MT)
C
      IF(MEF.EQ.0)GO TO 311
      IE=2
      DO 310 I=MEF,MEL
      IE=IE+1
      ZSYM(1,IE)=AMULTA*DSYM(1,MH,I)
      ZSYM(IE,1)=AMULTA*DSYM(1,I,MA)
      ZSYM(2,IE)=AMULTA*DSYM(1,MA,I)
      ZSYM(IE,2)=AMULTA*DSYM(1,I,MT)
      ZSYM(IE,IE)=CAY(I)
C
      IFF=2
```

```
      DO 310 J=MEF,MEL
      IFF=IEE+1
  310 ZSYM(IE,IFF)=ZSYM(IE,IEE)+AMULTA*DSYM(1,I,J)
C
  311 CONTINUE
C
C     FORM DO MATRIX
      IF(MJ.EQ.0) GO TO 340
C
      FSYM(1)=FSYM(1)-AMULTA*DSYM(1,MH,MJ)
      FSYM(2)=FSYM(2)-AMULTA*DSYM(1,MA,MJ)
      IE=2
      IF(MEF.EQ.0)GO TO 331
      DO 330 I=MEF,MEL
      IE=IE+1
  330 FSYM(IE)=FSYM(IE)-AMULTA*DSYM(1,I,MJ)
  331 CONTINUE
C
  340 CONTINUE
C
      IF(KMAN.EQ.0)GO TO 326
C
C     FOR NON ZERO QBAR,READ IN LOWEST K SYM AERO WITH NON 0 IMAG PARTS
C
      LOC=1+(IVBWN-1)*IBUMP
      CALL READMS(ND2,DSYM ,NWD,LOC)
C
C     VOBWN=1/K  FOR THIS AERO SET
C
C
      IF(KPRCHK.EQ.0)GO TO 572
      CALL PRNT(DSYM,NSYM,NSYM,NSYM,2,12HDSYMIMAGAERO   ,3)
  572 CONTINUE
C
C
C     CALC VELOCITY AERO
      AFACT=BZERO*SIZFCT*VOBWN /(12.*VTFPS)
C
C
C
      FSYM(1)=-AMULTA*(DSYM(2,MH,MA)+DSYM(2,MH,MH)*DELX)*QBAR*AFACT
     1 +FSYM(1)
      FSYM(2)=-AMULTA*(DSYM(2,MA,MA)+DSYM(2,MA,MH)*DELX)*QBAR*AFACT
     1 +FSYM(2)
C
      IF(MEF.EQ.0)GO TO 326
      IE=2
      DO 325 I=MEF,MEL
      IE=IE+1
  325 FSYM(IE)=-AMULTA*(DSYM(2,I,MA)+DSYM(2,I,MH)*DELX)*QBAR*AFACT
     1 +FSYM(IE)
C
```

```
  326 CONTINUE
C
C
C
      FSYM(1)=FSYM(1)+EMBAR(MH,MH)*ANGEE
      FSYM(2)=FSYM(2)+EMBAR(MA,MH)*ANGEE
      IE=2
      IF(MEF.EQ.0)GO TO 351
      DO 350 I=MEF,MEL
      IE=IE+1
  350 FSYM(IE)=FSYM(IE)+EMBAR(I,MH)*ANGEE
  351 CONTINUE
C
C     ADD THRUST TERMS
      IF(NENGS.EQ.0)GO TO 450
C     READ IN THRLOD AND THRGNF FROM UNIT DATASET
      CALL RDUNIT (4,0,THRLOD,NOUT,NER)
      CALL RDUNIT (5,0,THRGNF,NOUT,NER)
      DO 410 I=1,NENGS
      FSYM(1)=FSYM(1)+THRGNF(MH,I)*THRUST(I)
      FSYM(2)=FSYM(2)+THRGNF(MA,I)*THRUST(I)
      IE=2
      IF(MEF.EQ.0)GO TO 409
      DO 408 J=MEF,MEL
      IE=IE+1
  408 FSYM(IE) = FSYM(IE) + THRGNF(IE,I)*THRUST(I)
  409 CONTINUE
  410 CONTINUE
  450 CONTINUE
C
      IF(KPRCHK.EQ.0)GO TO 570
      CALL PRNT(ZSYM,NMS,NMS,NMS,1,4HZSYM    ,1)
      CALL PRNT(FSYM,NMS,1,NMS,1,4HFSYM    ,1)
      WRITE(NOUT,60)IVBW
  570 CONTINUE
C
      DET=0.
      CALL MIS1(ZSYM,NMS,NMS,FSYM,1,NER,DET)
      IF(NER.EQ.0)GO TO 680
      WRITE(NOUT,10) IVBW,VOBW
      GO TO 5000
C
  680 CONTINUE
      PALPH=FSYM(1)/(RBRADF*RAD)
      PDELT=FSYM(2)/(RBRADF*RAD)
C
      DO 690 I=1,NMODES
      QDOT(I)=0.
  690 Q(I)=0.
      Q(MA)=FSYM(1)
      QDOT(MA)=QBAR
      QDOT(MH)=DELX*QBAR
```

```
      Q(MT)=FSYM(2)
      IF(MJ.NE.0)Q(MJ)=1.
      IF(MEF.EQ.0)GO TO 696
      IE=2
      DO 695 I=MEF,MEL
      IE=IE+1
  695 Q(I)=FSYM(IE)
C
  696 CONTINUE
C
      LINES=LINES+4
      IF(LINES.LT.KROW)GO TO 880
      CALL HEADNG
      WRITE(NOUT,50)VKEAS,VELUN(IUNIT),ALT
      LINES=9
  880 CONTINUE
      WRITE(NOUT,40)
      DO 890 JJ=1,NSYM
      LINES=LINES+1
      IF(LINES.LT.KROW)GO TO 895
      CALL HEADNG
      WRITE(NOUT,50)VKEAS,VELUN(IUNIT),ALT
      WRITE(NOUT,40)
      LINES=9
  895 CONTINUE
      WRITE(NOUT,30)JJ,Q(JJ),QDOT(JJ)
  890 CONTINUE
C
      WRITE(NOUT,4)PALPH,PDELT
      LINES=LINES+2
C
C
C
      IF(INDSYM.EQ.1)GO TO 2001
C     READ IN LOWST K GENLZD ASYM AERO INTO DASYM
C
      NWD = 2*NASYM*NASYM
      LOC=2+(IVBW-1)*IBUMP
      CALL READMS(ND2,DASYM,NWD,LOC)
C
C     VCBW=1/K    FOR THIS AERO SET
C
C     FORM EQNS OF EQUILIBRIUM
C
      DO 1300 I=1,NMA
      FASYM(I)=0.
      DO 1300 J=1,NMA
 1300 ZASYM(I,J)=0.
C
C     FORM Z MATRIX
C
      MR=AMODNC(11)
```

```
      MY=AMODNO(12)
      ML=AMODNO(13)
      MJA=AMODNO(18)
      MEFA=AMODNO(14)
      MELA=AMODNO(15)
      MTR=AMODNO(16)
      MTY=AMODNO(17)
      LINES=LINES+18
      IF(LINES.GE.KROW)CALL HEADNG
      WRITE(NOUT,111)MR,ML,MY,MTR,MTY,MJA,MEFA,MELA,VTFPS,DYNP,AB,AC,
     1 PBARP,R3ARP,RTURN
       DELZ=EMBAR(MR,ML)/EMBAR(ML,ML)
C
C
      IF(KPRCHK.EQ.0)GO TO 573
      CALL  PRNT(DASYM,NASYM,NASYM,NASYM,2,12HASYMREALAERO    ,3)
  573 CONTINUE
C
      IF(MTR.EQ.0)GO TO 1015
      IF(MTY.EQ.0)GO TO 1015
C
      MR  =  MR  -  NSYM
      MY  =  MY  -  NSYM
      ML  =  ML  -  NSYM
      MJA=MJA  -  NSYM
      MEFA=MEFA-  NSYM
      MELA=MELA-NSYM
      MTR=MTR  -  NSYM
      MTY=MTY  -  NSYM
C
      ZASYM(1,1)=AMULTA*DASYM(1,ML,MY)
      ZASYM(1,2)=AMULTA*DASYM(1,ML,MTY)
      ZASYM(1,3)=AMULTA*DASYM(1,ML,MTR)
      ZASYM(2,1)=AMULTA*DASYM(1,MR,MY)
      ZASYM(2,2)=AMULTA*DASYM(1,MR,MTY)
      ZASYM(3,1)=AMULTA*DASYM(1,MY,MY)
      ZASYM(3,2)=AMULTA*DASYM(1,MY,MTY)
      ZASYM(3,3)=AMULTA*DASYM(1,MY,MTR)
      ZASYM(2,3)=AMULTA*DASYM(1,MR,MTR)
C
      IF(MEFA.LE.0)GO TO 1311
      IE=3
      DO 1310  I=MEFA,MELA
      IF=IE+1
      ZASYM(1,IF)=AMULTA*DASYM(1,ML,I)
      ZASYM(IF,1)=AMULTA*DASYM(1,I,MY)
      ZASYM(2,IF)=AMULTA*DASYM(1,MR,I)
      ZASYM(IF,2)=AMULTA*DASYM(1,I,MTY)
      ZASYM(3,IF)=AMULTA*DASYM(1,MY,I)
      ZASYM(IF,3)=AMULTA*DASYM(1,I,MTR)
      II=I+NSYM
      ZASYM(IF,IE)=CAY(II)
```

```
C
      IEE=3
      DO 1310 J=MEFA,MELA
      IEE=IEE+1
 1310 ZASYM(IE,IEE)=ZASYM(IE,IEE)+AMULTA*DASYM(1,I,J)
C
 1311  CONTINUE
C
C     FORM DO MATRIX
C
      IF(MJA.LE.0)GO TO 1340
C
      FASYM(1)=FASYM(1)-AMULTA*DASYM(1,ML,MJA)
      FASYM(2)=FASYM(2)-AMULTA*DASYM(1,MR,MJA)
      FASYM(3)=FASYM(3)-AMULTA*DASYM(1,MY,MJA)
      IE=3
      IF(MEFA.EQ.0)GO TO 1331
      DO 1330 I=MEFA,MELA
      IE=IE+1
 1330  FASYM(IE)=FASYM(IE)-AMULTA*DASYM(1,I,MJA)
 1331  CONTINUE
C
 1340 CONTINUE
C
C
C     READ IN LOWEST K ASYM AERO WITH NON 0 IMAG PARTS
C
C     VOBWN=1/K  FOR THIS AERO SET
C
      LOC=2+(IVBWN-1)*IBUMP
      CALL READMS(ND2,DASYM,NWD,LOC)
C
      IF(KPRCHK.EQ.0)GO TO 574
C
      CALL PRNT(DASYM,NASYM,NASYM,NASYM,2,12HASYMIMAGAERO    ,3)
  574 CONTINUE
C
C     CALC VELOCITY AERO
C
      AFACTA=BZERO*SIZFCT*VOBWN /(12.*VTFPS)
C
C
C
      FASYM(1)=-AMULTA*(DASYM(2,ML,MR)+DASYM(2,ML,ML)*DELZ)*PBAR*AFACTA
     1 -AMULTA*(DASYM(2,ML,MY)-DASYM(2,ML,ML)*DELX)*RBAR*AFACTA
     1 +FASYM(1)
      FASYM(2)=-AMULTA*(DASYM(2,MR,MR)+DASYM(2,MR,ML)*DELZ)*PBAR*AFACTA
     1 -AMULTA*(DASYM(2,MR,MY)-DASYM(2,MR,ML)*DELX)*RBAR*AFACTA
     1 +FASYM(2)
      FASYM(3)=-AMULTA*(DASYM(2,MY,MR)+DASYM(2,MY,ML)*DELZ)*PBAR*AFACTA
     1 -AMULTA*(DASYM(2,MY,MY)-DASYM(2,MY,ML)*DELX)*RBAR*AFACTA
     1 +FASYM(3)
```

```
C
      IF(MEFA.LE.0)GO TO 1326
      IF=3
      DO 1325 I=MEFA,MELA
      IF=IF+1
 1325 FASYM(IF)=-AMULTA*(DASYM(2,I,MR)+DASYM(2,I,ML)*DELZ)*PBAR*AFACTA
     1 -AMULTA*(DASYM(2,I,MY)-DASYM(2,I,ML)*DELX)*RBAR*AFACTA
     1 +FASYM(IF)
C
 1326  CONTINUE
C
C
C
C
C
      IF(KPRCHK.EQ.0)GO TO 1570
      CALL  PRNT(ZASYM,NMA,NMA,NMA,1,8H ZASYM     ,2)
      CALL  PRNT(FASYM,NMA,1,NMA,1,8H FASYM     ,2)
      WRITE(NOUT,60)IVBW
 1570 CONTINUE
C
      DET=0.
      CALL  MISL(ZASYM,NMA,NMA,FASYM,1,NER,DET)
      IF(NER.EQ.0)GO TO 1680
      WRITE(NOUT,10)  IVBW,VOBW
C
 1680 CONTINUE
      PBETA=FASYM(1)/(RBRADF*RAD)
      PDELY=FASYM(2)/(RBRADF*RAD)
      PDELR=FASYM(3)/(RBRADF*RAD)
C
      MR  =  MR  +  NSYM
      MY  =  MY  +  NSYM
      ML  =  ML  +  NSYM
      MJA=MJA  +  NSYM
      MEFA=MEFA+  NSYM
      MELA=MELA+NSYM
      MTY=MTY  +  NSYM
      MTR=MTR  +  NSYM
      Q(MY)=FASYM(1)
      Q(MTY)=FASYM(2)
       Q(MTR)=FASYM(3)
       QDOT(MR)=PBAR
      QDOT(ML)=DELZ*PBAR  -DELX*RBAR
      QDOT(MY)=RBAR
      IF(MJA.NE.0)Q(MJA)=1.
      IF(MEFA.EQ.0)GO TO 1696
      IF=3
      DO 1695  I=MEFA,MELA
      IF=IF+1
 1695  Q(I)=FASYM(IF)
C
```

```
 1696  CONTINUE
C
       LINES=LINES+2
       IF(LINES.LT.KROW)GO TO 1880
       CALL HEADNG
       WRITE(NOUT,50)VKFAS,VELUN(IUNIT),ALT
       LINES=9
 1880  CONTINUE
       WRITE(NOUT,40)
       DO 1890 J=1,NASYM
       JJ=J+NSYM
       LINES=LINES+1
       IF(LINES.LT.KROW)GO TO 1895
       CALL HEADNG
       WRITE(NOUT,50)VKFAS,VELUN(IUNIT),ALT
       WRITE(NOUT,40)
       LINES=9
 1895   CONTINUE
       WRITE(NOUT,30)JJ,Q(JJ),QDOT(JJ)
 1890  CONTINUE
C
       WRITE(NOUT,5)PBETA,PDELY,PDELR
       LINES=LINES+3
C
 2001  CONTINUE
C
C      READ IN PIQ
C
       CALL RDUNIT (6,0,PIQ,NOUT,NER)
       IF (NER.NE.0) GO TO 5000
       IF(KPRCHK.EQ.0)GO TO 108
       CALL PRNT (PIQ,NINTLD,NMODES,NINTLD,1,4HPIQ ,1)
C
  108  CONTINUE
C
       DO 110 I=1,NINTLD
       PS(I)=0.
       PA(I)=0.
       PS(I)=PS(I)+PIQ(I,MH)*ANGEE
       IF(NENGS.EQ.0)GO TO 110
       DO 109 J=1,NENGS
  109  PS(I)=PS(I)+THPLOD(I,J)*THRUST(J)
 110   CONTINUE
C
C      CALC DYNP X MOTION DEPENDENT AERO FORCES X Q
C
C
C      READ IN PAQS(L) AND PAQA(L) FOR K=0  (VOBWSM AND VOBWAM)
C
       CALL RDUNIT (7,IVBW,PAQS,NOUT,NER)
       IF(INDSYM.EQ.1)GO TO 112
       CALL RDUNIT (8,IVBW,PAQA,NOUT,NER)
```

```
  112 CONTINUE
      IF (NER.NE.0) GO TO 5000
C
      IF(KPRCHK.EQ.0)GO TO 115
      CALL PRNT(PAQS,NINTLD,NSYM,NINTLD,2,12HPAQSREALAERO    ,3)
      IF(INDSYM.EQ.1)GO TO 115
      CALL PRNT(PAQA,NINTLD,NASYM,NINTLD,2,12HPAQAREALAERO    ,3)
  115 CONTINUE
C
C
      AMULT=DYNP
      DO 130 I=1,NINTLD
C
      DO 120 K=1,NSYM
  120 PS(I)=PS(I)+PAQS(1,I,K)*Q(K)*AMULT
      IF (INDSYM.EQ.1) GO TO 130
      DO 125 K=1,NASYM
      KK=K+NSYM
  125 PA(I)=PA(I)+PAQA(1,I,K)*Q(KK)*AMULT
  130 CONTINUE
C
      IF(KMAN.EQ.0)GO TO 210
C
C     READ IN PACS FOR VOBWN  AND PAQA FOR VOBWN
C
      CALL RDUNIT (7,IVBWN,PACS,NOUT,NER)
      IF(INDSYM.EQ.1)GO TO 117
      CALL RDUNIT (8,IVBWN,PAQA,NOUT,NER)
  117 CONTINUE
C
      IF(KPRCHK.EQ.0)GO TO 116
      CALL PRNT(PAQS,NINTLD,NSYM,NINTLD,2,12HPAQSIMAGAERO    ,3)
      IF(INDSYM.EQ.1)GO TO 116
      CALL PRNT(PAQA,NINTLD,NASYM,NINTLD,2,12HPAQAIMAGAERO    ,3)
  116 CONTINUE
C
C
      DO 200 I=1,NINTLD
      DO 180 K=1,NSYM
  180 PS(I)=PS(I)+PAQS(2,I,K)*QDOT(K)*AMULT*AFACT
      IF (INDSYM.EQ.1) GO TO 200
      DO 190 K=1,NASYM
      KK=K+NSYM
  190 PA(I)=PA(I)+PAQA(2,I,K)*QDOT(KK)*AMULT*AFACTA
C
  200 CONTINUE
C
C
  210 CONTINUE
C     PRINT LOADS IF DESIRED
C
      CALL HEADNG
```

```
      WRITE(NOUT,2)
    2 FORMAT (1H0,20X,*SYM INTEGRATED LOADS FOR TRIM*)
      CALL PRNT(PS,NINTLD,1,NINTLD,1,4H        ,1)
      IF (INDSYM.EQ.1) GO TO 280
      WRITE (NOUT,3)
    3 FORMAT (1H0,20X,*ASYM INTEGRATED LOADS FOR TRIM*)
      CALL PRNT(PA,NINTLD,1,NINTLD,1,4H        ,1)
C
  280 CONTINUE
C
C
C     NOW SAVE PS AND PA FOR SUBSEQUENT USE
C
C
 5000 RETURN
C
 1015 CONTINUE
      WRITE(NOUT,1016)
 1016 FORMAT(1H1,20X,*CANNCT TRIM,NO TRIM MODE SPECIFIED*)
      STOP
C
 1017 CONTINUE
      WRITE(NOUT,1018)AN
 1018 FORMAT(1H1,20X, HEY TRY SOME LOAD FACTOR OTHER THAN  ,F8.4,
     1   FOR THE TURN )
      STOP
C
      END
```

```
      SUBROUTINE ACCHST                                                 ACCHST 2
     1 ( FTG,NACC,NFREQ,NTMRSP,OMEGMX, INDSYM,OMEGA                     ACCHST 3
     2 , AS,AA,AST,AAT,TIME,TIMEMX,KPRBLS,NG,NGL, IPLBL,PLOT)           ACCHST 4
C                                                                       ACCHST 5
      COMMON /ZZZ/CASE(48),NIN,NOUT,KROW,LINES, IPRNT,NFR               ACCHST 6
C                                                                       ACCHST 7
      DIMENSION FTG(2,1)                                                ACCHST 8
      DIMENSION OMEGA(1), TIME(1), PLOT(1)                              ACCHST 9
      DIMENSION AS(2,NACC,NFREQ), AA(2,NACC,NFREQ)                      ACCHST10
      DIMENSION AST(NACC,1), AAT(NACC,1)                                ACCHST11
C                                                                       ACCHST12
      DATA TWOPI/6.2831853/                                             ACCHST13
C                                                                       ACCHST14
   10 FORMAT (1H0,20X,*SYMMETRIC TIME RESPONSE - INCREMENTAL ACCELERATIOACCHST15
     1NS*)                                                              ACCHST16
   11 FORMAT (1H0,20X,*UNSYMMETRIC TIME RESPONSE - INCREMENTAL ACCELERATACCHST17
     1IONS*)                                                            ACCHST18
   12 FORMAT (1H0,20X,F8.4,* RESPONSE SECS*,2X,F8.4,* HZ COUNTED*)      ACCHST19
   40 FORMAT (1H0,2X,*TIME  *,8(7X ,*STA*, I3))                         ACCHST20
   50 FORMAT (1H ,F8.4,8(1X,E12.5))                                     ACCHST21
C                                                                       ACCHST22
C     INDSYM=1 IS A SYMMETRIC CASE, INDSYM=2, UNSYMMETRIC               ACCHST23
C                                                                       ACCHST24
      DO 100 J=1,NFREQ                                                  ACCHST25
      AR = FTG(1,J)                                                     ACCHST26
      AI = FTG(2,J)                                                     ACCHST27
      DO 100 I=1,NACC                                                   ACCHST28
      T = AS(1,I,J)*AR - AS(2,I,J)*AI                                   ACCHST29
      AS(2,I,J) = AS(1,I,J)*AI + AS(2,I,J)*AR                           ACCHST30
      AS(1,I,J) = T                                                     ACCHST31
      IF (INDSYM.EQ.1) GO TO 100                                        ACCHST32
      T = AA(1,I,J)*AR - AA(2,I,J)*AI                                   ACCHST33
      AA(2,I,J) = AA(1,I,J)*AI + AA(2,I,J)*AR                           ACCHST34
      AA(1,I,J) = T                                                     ACCHST35
  100 CONTINUE                                                          ACCHST36
C                                                                       ACCHST37
      CALL IFT (NFREQ,OMEGA,OMEGMX,0.,0,NACC,AS,NTMRSP,TIME,AST,TIMEMX) ACCHST38
C                                                                       ACCHST39
      IF (INDSYM.EQ.1) GO TO 150                                        ACCHST40
      CALL IFT (NFREQ,OMEGA,OMEGMX,0.,0,NACC,AA,NTMRSP,TIME,AAT,TIMEMX) ACCHST41
C                                                                       ACCHST42
  150 CONTINUE                                                          ACCHST43
C                                                                       ACCHST44
      AMA = 1.0                                                         ACCHST45
      IF (NGL.LT.0) AMA = -1.0                                          ACCHST46
C                                                                       ACCHST47
      DO 500 I=1,NACC                                                   ACCHST48
C                                                                       ACCHST49
      IF (INDSYM.EQ.1) GO TO 500                                        ACCHST50
C                                                                       ACCHST51
C     CALC LHS AND RHS ACC. (UNSYMMETRIC LOAD CONDITION)                ACCHST52
C                                                                       ACCHST53
```

```
      DO 200 J=1,NTMRSP                                                  ACCHST54
      TEMP = AST(I,J) - AMA*AAT(I,J)                                    ACCHST55
      AST(I,J) = AST(I,J) + AMA*AAT(I,J)                                ACCHST56
  200 AAT(I,J) = TEMP                                                   ACCHST57
C                                                                       ACCHST58
  500 CONTINUE                                                          ACCHST59
C                                                                       ACCHST60
      IF (IPLBL.EQ.0) GO TO 800                                         ACCHST61
C                                                                       ACCHST62
C     NEW PLOT ACCELERATION TIME HISTORIES                              ACCHST63
C                                                                       ACCHST64
      DO 750 I=1,NACC                                                   ACCHST65
      IF (IPLBL.EQ.2) GO TO 720                                         ACCHST66
C                                                                       ACCHST67
      DO 710 J=1,NTMRSP                                                 ACCHST68
  710 PLOT(J) = AST(I,J)                                                ACCHST69
      CALL PLT1 (NTMRSP,TIME,PLOT,1)                                    ACCHST70
      WRITE (NOUT,21) I,NG                                              ACCHST71
   21 FORMAT (1H0,20X,*ACCELERATION TIME HISTORY RHS ACC. NO.*,I4,4X    ACCHST72
     1 ,*AND ORIENTATION NO.*,I4)                                       ACCHST73
C                                                                       ACCHST74
  720 IF (IPLBL.EQ.1) GO TO 750                                         ACCHST75
      IF (INDSYM.EQ.1) GO TO 750                                        ACCHST76
C                                                                       ACCHST77
      DO 730 J=1,NTMRSP                                                 ACCHST78
  730 PLOT(J) = AAT(I,J)                                                ACCHST79
      CALL PLT1 (NTMRSP,TIME,PLOT,1)                                    ACCHST80
      WRITE (NOUT,22) I,NG                                              ACCHST81
   22 FORMAT (1H0,20X,*ACCELERATION TIME HISTORY LHS ACC. NO.*,I4,4X    ACCHST82
     1 ,*AND ORIENTATION NO.*,I4)                                       ACCHST83
C                                                                       ACCHST84
  750 CONTINUE                                                          ACCHST85
C                                                                       ACCHST86
  800 IF (KPRBLS.EQ.0) GO TO 1000                                       ACCHST87
C                                                                       ACCHST88
      FRQMX = OMEGMX/TWOPI                                              ACCHST89
      LINES = KROW+1                                                    ACCHST90
C                                                                       ACCHST91
      DO 820 JS=1,NACC,8                                                ACCHST92
      JF = MINO(JS+7,NACC)                                              ACCHST93
      IF (LINES+3.LE.KROW) GO TO 810                                    ACCHST94
      CALL HEADNG                                                       ACCHST95
      WRITE (NOUT,10)                                                   ACCHST96
      WRITE (NOUT,12) TIMEMX,FRQMX                                      ACCHST97
      LINES = LINES+4                                                   ACCHST98
  810 WRITE (NOUT,40) (J,J=JS,JF)                                       ACCHST99
      LINES = LINES+2                                                   ACCHS100
      DO 820 K=1,NTMRSP                                                 ACCHS101
      LINES = LINES+1                                                   ACCHS102
      IF (LINES.LE.KROW) GO TO 820                                      ACCHS103
      CALL HEADNG                                                       ACCHS104
      WRITE (NOUT,40) (J,J=JS,JF)                                       ACCHS105
```

```
      LINES = LINES+3                                          ACCHS106
  820 WRITE (NOUT,50) TIME(K), (AST(J,K),J=JS,JF)              ACCHS107
C                                                              ACCHS108
      IF (INDSYM.EQ.1) GO TO 1000                              ACCHS109
C                                                              ACCHS110
      DO 920 JS=1,NACC,8                                       ACCHS111
      JF = MINO(JS+7,NACC)                                     ACCHS112
      IF (LINES+3.LE.KROW) GO TO 910                           ACCHS113
      CALL HEADNG                                              ACCHS114
      WRITE (NOUT,11)                                          ACCHS115
      WRITE (NOUT,12) TIMEMX,FROMX                             ACCHS116
      LINES = LINES+4                                          ACCHS117
  910 WRITE (NOUT,40) (J,J=JS,JF)                              ACCHS118
      LINES = LINES+2                                          ACCHS119
      DO 920 K=1,NTMRSP                                        ACCHS120
      LINES = LINES+1                                          ACCHS121
      IF (LINES.LE.KROW) GO TO 920                             ACCHS122
      CALL HEADNG                                              ACCHS123
      WRITE (NOUT,40) (J,J=JS,JF)                              ACCHS124
      LINES = LINES+3                                          ACCHS125
  920 WRITE (NOUT,50) TIME(K), (AAT(J,K),J=JS,JF)              ACCHS126
C                                                              ACCHS127
 1000 RETURN                                                   ACCHS128
      END                                                      ACCHS129
```

```
      SUBROUTINE CGUST (CR,AMODNO,EMBAR,CAY)
C
      DIMENSION AMODNO(1),EMBAR(1),CAY(1),CR(1)
C
      COMMON NAA, A(1)
C
      COMMON /ZZZ/HEDR(48),NIN,NOUT,KROW,LINES,IPRNT,NER
      COMMON /XTF/NF(100)
      EQUIVALENCE (NF(9),NFREC),     (NF(32),NINTLD)
      EQUIVALENCE (NF(51),AN),       (NF(52),ZDOT),     (NF(53),RTURN)
      EQUIVALENCE (NF(61),NORMAX),   (NF(62),TIMEMX),   (NF(63),EFR)
     1 , (NF(33),NSTRSS)
     1 , (NF(10),NENGS)
     1 , (NF(66),HBGRD)
     1 , (NF(43),NBOX),(NF(44),NSBETO)
     1 , (NF(97),DELT)
     1 , (NF(29),NACC)
C
      DATA CORERQ/8H BLAST  /
C
      L1   = NAA
      L2   = L1   + NINTLD
      L21  = L2 + NINTLD
      L22  = L21  + NSTRSS*NINTLD
      L23  = L22  + NENGS
      L24  = L23  + NINTLD
      NAA  = L24  + NINTLD
C
C     READ STRESS MATRIX FROM UNIT DATASET IF NEEDED
      IF(NSTRSS.NE.0)CALL RDUNIT(3,0,A(L21),NOUT,NER)
C
C     READ IN SYMCOD FROM UNIT LOAD TAPE
      CALL RDUNIT(9,0,A(L23),NOUT,NER)
C     READ IN ASMCOD FROM UNIT LOAD TAPE
      CALL RDUNIT(10,0,A(L24),NOUT,NER)
C
C
C     NOW READ IN ENGINE THRUSTS IF ANY DEFINED
C
      IF(NENGS.EQ.0)GO TO 100
      I1=L22
      I2=L22+NENGS-1
C     READ IN THRUST
      READ(NIN,310) (A(I),I=I1,I2)
  100 CONTINUE
C
      CALL CTRIM(AMODNO,EMBAR,CAY,A(L1),A(L2),A(L22))
C
      L3   = NAA
      L4   = L3   + NORMAX
      L5   = L4   + NORMAX
      L6   = L5   + NINTLD*8
```

```
      L7   = L6   + NFREQ                                               CGUST  54
C                                                                       CGUST  55
      I1  =  L3                                                         CGUST  56
      I2  =  L3  + NORMAX-1                                             CGUST  57
C     READ  IN ORIENTATIONS                                             CGUST  58
      READ  (NIN,300)  (A(I),I=I1,I2)                                   CGUST  59
      I1  =  L4                                                         CGUST  60
      I2  =  L4  + NORMAX-1                                             CGUST  61
C     READ  IN RANGES                                                   CGUST  62
      READ  (NIN,310)  (A(I),I=I1,I2)                                   CGUST  63
C                                                                       CGUST  64
C                                                                       CGUST  65
C     READ  IN STALDS FROM LOAD DATASET                                 CGUST  66
      CALL  RDLOAD (2,0,A(L5),NOUT,NER)                                 CGUST  67
C     READ  CR FROM LOAD DATASET                                        CGUST  68
      CALL  RDLOAD (3,0,CR,NOUT,NER)                                    CGUST  69
C     READ  OMEGA FROM LOAD DATASET                                     CGUST  70
      CALL  RDLOAD (4,0,A(L6),NOUT,NER)                                 CGUST  71
C                                                                       CGUST  72
      NTMRSP=0                                                          CGUST  73
      NT=TIMEMX/DELT+1                                                  CGUST  74
      DELTT=DELT                                                        CGUST  75
      IND=0                                                             CGUST  76
      TI=0.                                                             CGUST  77
      DO 200 I=1,NT                                                     CGUST  78
      TI=TI+DELTT                                                       CGUST  79
      IF(TI.GT.TIMEMX)GO TO 250                                         CGUST  80
      NTMRSP=NTMRSP+1                                                   CGUST  81
      IF(IND.GT.1)GO TO 200                                             CGUST  82
      IF(TI.LT.0.25)GO TO 200                                           CGUST  83
      IF(TI.LE.1.0)GO TO 150                                            CGUST  84
      IND=2                                                             CGUST  85
      TI=TI-DELTT                                                       CGUST  86
      DELTT=DELTT*2.0                                                   CGUST  87
      TI=TI+DELTT                                                       CGUST  88
      GO TO 200                                                         CGUST  89
  150 IF(IND.EQ.1)GO TO 200                                             CGUST  90
      IND=1                                                             CGUST  91
      TI=TI-DELTT                                                       CGUST  92
      DELTT=DELTT*5.0                                                   CGUST  93
      TI=TI+DELTT                                                       CGUST  94
  200 CONTINUE                                                          CGUST  95
  250 CONTINUE                                                          CGUST  96
      NTMRSP=NTMRSP+1                                                   CGUST  97
      IF(TI-DELTT.LT.TIMEMX.AND.ABS(TI-DELTT-TIMEMX).GT.0.010)          CGUST  98
     1 NTMRSP=NTMRSP+1                                                  CGUST  99
C                                                                       CGUST100
      NTMCST=400                                                        CGUST101
C                                                                       CGUST102
      L8  =  L7  +  MAX0(2*NINTLD*NFREQ,NSTRSS*NTMRSP,6*NBOX)           CGUST103
      L8  =  MAX0(L8,L7+2*NACC*NFREQ)                                   CGUST104
      L9  =  L8  +  MAX0(2*NINTLD*NFREQ,NSTRSS*NTMRSP,6*NSBETO)         CGUST105
```

```
      L9 = MAXO(L9,L8+2*NACC*NFREQ)                                     CGUST106
      L10 = L9  + NTMGST                                                CGUST107
      L11 = L10 + MAXO(NTMGST,NTMRSP)                                   CGUST108
      L12 = L11 + NTMGST                                                CGUST109
      L13 = L12 + NFREQ                                                 CGUST110
      L14 = L13 + NFREQ                                                 CGUST111
      L15 = L14 + 2*NFREQ                                               CGUST112
      L16 = L15 + NINTLD*NTMRSP                                         CGUST113
      L16 = MAXO(L16,L15+NACC*NTMRSP)                                   CGUST114
      L17 = L16 + NINTLD*NTMRSP                                         CGUST115
      L17 = MAXO(L17,L16+NACC*NTMRSP)                                   CGUST116
      L18 = L17 + NTMRSP                                                CGUST117
      L19 = L18 + MAXO(NINTLD*8,NSTRSS*8)                               CGUST118
      LMAX = L19 + 2*NINTLD                                             CGUST119
C                                                                       CGUST120
      WRITE(NOUT,999)LMAX,CORERQ                                        CGUST121
  999 FORMAT(1H0,20X,I10,  WORDS OF CORE RQD FOR STEP +++ ,A10, +++ )   CGUST122
C                                                                       CGUST123
      CALL GUSTDR(CR,NTMGST                                             CGUST124
     1 , A(L1),A(L2),A(L3),A(L4)                                        CGUST125
     2 , A(L9),A(L10),A(L11),A(L6),A(L12),A(L13)                        CGUST126
     2 , A(L7),A(L8),A(L15),A(L16)                                      CGUST127
     3 , A(L7),A(L8),A(L15),A(L16),A(L17),A(L18),A(L5),NTMRSP,          CGUST128
     4 A(L21),A(L7),A(L8),A(L19),A(L23),A(L24),NINTLD,A(L14),           CGUST129
     5 A(L7),A(L8))                                                     CGUST130
C                                                                       CGUST131
  300 FORMAT (6I12)                                                     CGUST132
  310 FORMAT (6F12.0)                                                   CGUST133
C                                                                       CGUST134
      RETURN                                                            CGUST135
      END                                                               CGUST136
```

```
      SUBROUTINE FLTPOS(
     1 T,SLNTRC,VEL,ALT,AN,AB,AC,GAMX,GAMY,GAMZ,KMAN,SLNTRG,XB,YB,ZB,
     1 LBURST,XF,YF,ZF)
C
C
C     FLTPOS CALCULATES AIRCRAFT AND BURST POS IN EFAS AT TIME T
C     ASSUMING THAT THE MEAN PATH FLOWN IS AS DEFINED BY THE MANEUVER
C
C     IF LBURST=0  DO NOT CALC OR PRINT BURST POSITION
C     IF LBURST=1  DO     CALC AND PRINT BURST POSITION
C
      COMMON/ZZZ/CASE(48),NIN,NOUT,KROW,LINES,IPRNT,NER
C
      COMMON/XTF/NF(100)
C
      EQUIVALENCE  (NF(66),HGRD)
C
      DATA GEE/32.174/
      DATA RAD/0.017453/
C
      SLNTRC=-SLNTRC
      IF  (KMAN.EQ.0) GO TO 300
      IF  (KMAN.EQ.1) GO TO 200
C
C     CLIMBING TURN
C
      ANGVEL=GEE*TAN(RAD*AB)/VEL
      RADIUS=VEL/ANGVEL
C
      YF=RADIUS*COS(ANGVEL*T)
      XF=-RADIUS*SIN(ANGVEL*T)
      ZF=ALT+VEL*SIN(RAD*AC)*T
C
      IF(LBURST.EQ.0)GO TO 1000
      XB=SLNTRC*(COS(RAD*AC)*GAMX+SIN(RAD*AB)*SIN(RAD*AC)*GAMY
     1 +COS(RAD*AB)*SIN(RAD*AC)*GAMZ)
      YB=SLNTRC*(COS(RAD*AB)*GAMY-SIN(RAD*AB)*GAMZ)+RADIUS
      ZB=SLNTRC*(-SIN(RAD*AC)*GAMX+SIN(RAD*AB)*COS(RAD*AC)*GAMY
     1 +COS(RAD*AB)*COS(RAD*AC)*GAMZ)+ALT
C
      GO TO 1000
C
  200 CONTINUE
C
C     PULLOUT
C
      ANGVEL=(AN-1.0)*GEE/VEL
      RADIUS=VEL/ANGVEL
C
      XF=-RADIUS*SIN(ANGVEL*T)
      YF=0.
      ZF=ALT+RADIUS*(1.0-COS(ANGVEL*T))
```

```
C
      IF(LBURST.EQ.0)GO TO 1000
      XB=SLNTRO*GAMX
      YB=SLNTRO*GAMY
      ZB=SLNTRO*GAMZ+ALT
C
      GO TO 1000
C
  300 CONTINUE
C
C     LEVEL FLIGHT
C
      XE=-VEL*T
      YE=0.
      ZE=ALT
C
      IF(LBURST.EQ.0)GO TO 1000
      XB=SLNTRO*GAMX
      YB=SLNTRO*GAMY
      ZB=SLNTRO*GAMZ+ALT
C
      GO TO 1000
C
 1000 CONTINUE
C
      SLNTRG=SQRT((XE-XB)**2+(YE-YB)**2+(ZE-ZB)**2)
C
      SLNTRO=-SLNTRG
C
      DX=XE-XB
      DY=YE-YB
      DZ=ZE-ZB
C
C
      IF(LBURST.NE.0)
     1WRITE(NOUT,1)T,XE,YE,ZE,XB,YB,ZB,
     1 SLNTRO,SLNTRG,DX,DY,DZ
C
    1 FORMAT(1H0,20X, COORDINATES OF AIRCRAFT AND BURST AT TIME =  ,
     1 F10.4,  SECS /1H0,
     1 1H0,20X, AIRCRAFT (EFAS) /
     1 1H ,20X, X =   ,E12.5/
     1 1H ,20X, Y =   ,E12.5/
     1 1H ,20X, Z =   ,E12.5/
     1 1H0,20X, BURST (EFAS) /
     1 1H ,20X, X =   ,E12.5/
     1 1H ,20X, Y =   ,E12.5/
     1 1H ,20X, Z =   ,E12.5/
     1 1H0,20X, DISTANCE BURST TO AIRCRAFT AT INTERCEPT /
     1 1H ,20X, SLNTRO =   ,E12.5,  FT. /
     1 1H0,20X, DISTANCE BURST TO AIRCRAFT NOW IS /
     1 1H ,20X, SLNTRG =   ,E12.5,  FT. /
```

```
     1 1H ,20X, XE-XB  =   ,E12.5,   FT. /                                FLTP0106
     1 1H ,20X, YE-YB  =   ,E12.5,   FT. /                                FLTP0107
     1 1H ,20X, ZE-ZB  =   ,E12.5,   FT. )                                FLTP0108
C                                                                         FLTP0109
      IF(LBURST.EQ.0)GO TO 2000                                           FLTP0110
      IF(ZB.GE.HGRD)GO TO 2000                                            FLTP0111
C                                                                         FLTP0112
      ZBBG=HGRD-ZB                                                        FLTP0113
      WRITE(NOUT,2)ZBBG                                                   FLTP0114
    2 FORMAT(1H0,20X, *** ATTENTION *** BURST IS   ,F10.2,                FLTP0115
     1   FEET BELOW GROUND LEVEL )                                        FLTP0116
C                                                                         FLTP0117
 2000 CONTINUE                                                            FLTP0118
C                                                                         FLTP0119
      RETURN                                                              FLTP0120
      END                                                                 FLTP0121
```

```
      SUBROUTINE GSTHST(
     1 RHOA,NTMGST,DELTG,SCALT,VELG,TIMG,RHO,VSS,DELPR,VGUST,
     1 SCALR,PO,EFR,VEL,SLNTRO,ALT,TARR,AN,AB,AC,GAMX,GAMY,GAMZ,KMAN,
     1 KGRD,HGRD,XB,YB,ZB,XE,YE,ZE)
C
      COMMON/ZZZ/CASE(48),NIN,NOUT,KROW,LINES,IPRNT,NER
C
      COMMON/XTF/NF(100)
      EQUIVALENCE (NF(22),KPRTMH)
     1 , (NF(62),TIMEMX)
     2 , (NF(89),TAU)
     3 , (NF(1),AMACH),(NF(14),SIGMA),(NF(13),RHO0)
C
C
      DIMENSION VELG(1),TIMG(1),RHO(1)
      DIMENSION SHBT(14),CSGOTP(14),CCTP(16),SHBTA(16)
C
      DATA SHBT/0.,0.1,0.2,0.4,0.6,0.8,1.0,1.2,1.4,1.6,1.8,2.0,2.2,2.4/
      DATA CSGOTP/0.,0.085,0.165,0.35,0.6,0.98,1.55,2.45,3.6,5.0,6.8,
     1 8.9,11.2,14./
      DATA CCTP/1.42,0.5,0.323,0.2,0.144,0.1075,0.083,0.066,0.054,0.047,
     1 0.043,0.041,0.040,0.040,0.,0./
      DATA SHBTA/0.025,0.05,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1.0,
     1 1.1,1.2,0.,0./
C
C
      RHOA=RHO0*SIGMA
      IF(KPRTMH.EQ.0)GO TO 20
      WRITE(NOUT,3)ALT,VEL,SIGMA,RHOA,EFR,SLNTRO
    3 FORMAT(1H0, ALT= ,E12.5,  FT. /1H ,
     1  VEL= ,E12.5,  FPS /1H ,
     1  SIGMA= ,E12.5/1H ,
     1  RHO= ,E12.5, LB-SEC2/FT4 /1H ,
     1  EFR= ,E12.5, KILOTONS /1H ,
     1  SLNTRO= ,E12.5,  FT. )
C
   20 CONTINUE
C
      TEMP=(EFR*14.696/PO)**(1./3.)
      SCALR=0.001/TEMP
      SCALT=1115.2*TEMP/VSS
C
C     NOW SET UP FOR POSSIBLE TPP CALCS
C
      KGRDT=KGRD
      IF(KGRDT.EQ.0)GO TO 50
      SHB=SCALR*(ZB-HGRD)
      IF(SHB.GT.0.025)GO TO 30
      KGRDT=0
      SCALR=SCALR/1.2
      SCALT=SCALT*1.2
      GO TO 50
```

```
C
   30 CONTINUE
      IF(SHB.LT.2.4)GO TO 35
      SGOTP=1.569*SHB**2.5
      GO TO 40
C
   35 CONTINUE
      CALL INTRPB(SHB,0.,0.,14,0,0,SHBT,0.,0.,CSGOTP,SGOTP)
   40 CONTINUE
      SH=AMIN1(SHB,1.1)
      CALL INTRPB(SH,0.,0.,16,0,0,SHBTA,0.,0.,CCTP,CTP)
      SRGOTP=SQRT(SCCTP**2+SHB**2)
C
   50 CONTINUE
C
      RTA=SCALR*SLNTRO
      RR=1.
      IF(RTA.GT.0.55)RR=1.0+0.06*(RTA-0.55)**(1./3.)
      RP=RTA*RR
C
      CALL PRESS(RP,ZETA)
      DELPR=ZETA
      DELP=ZETA*PO
      TARR=0.
      CALL TAR(RTA,TARR)
      TARR=-TARR*SCALT
      IF(KPRTMH.EQ.0)GO TO 60
      WRITE(NOUT,5)ZETA,AMACH,RTA,PO,DELP,SCALR,SCALT
    5 FORMAT(1H0, ZETA= ,E12.5/1H ,
     1  MACH= ,F8.4/1H ,
     1  RTA= ,E16.5/1H ,
     1  PO= ,F12.5,  PSI /1H ,
     1  DELP= ,F12.5,  PSI /1H ,
     1  SCALR= ,E12.5/1H ,
     1  SCALT= ,E12.5)
    4 FORMAT(1H0,20X,
     1  TIME BACK TO BURST =  ,F10.4,  SECS )
C
   60 CONTINUE
      WRITE(NOUT,4)TARR
C
      INO=0
      ICNT=0
C
      T=0.0
C
      IF(KPRTMH.EQ.0)GO TO 90
      CALL HEADNG
C
      WRITE(NOUT,2)
    2 FORMAT(1H),10X, GUST TIME HISTORY AT AAS ORIGIN /1H0,
     1  ITM ,5X, TIME ,2X, VELOCITY ,6X, DENSITY ,7X, SLNTRG )
```

```
C                                                                        GSTHS106
   90 CONTINUE                                                           GSTHS107
C                                                                        GSTHS108
      DO 100 I=1,NTMGST                                                  GSTHS109
      ICNT=ICNT+1                                                        GSTHS110
C                                                                        GSTHS111
      TIMG(I) = T-TAU                                                    GSTHS112
C                                                                        GSTHS113
      CALL FLTPOS(T,SLNTRO,VEL,ALT,AN,AB,AC,GAMX,GAMY,GAMZ,KMAN,         GSTHS114
     1 SLNTRG,XB,YB,ZB,O,XE,YE,ZE)                                       GSTHS115
C                                                                        GSTHS116
      IF(KGRDT.EQ.0)GO TO 80                                             GSTHS117
      SHA=SCALR*(ZE-HGRD)                                                GSTHS118
      SCA=SCALR*SQRT((XE-XB)**2+(YE-YB)**2)                              GSTHS119
   80 CONTINUE                                                           GSTHS120
C                                                                        GSTHS121
      TIMTOT=T-TARP                                                      GSTHS122
C                                                                        GSTHS123
      CALL TPEVAL(TIMTOT,RHCA,VSS,SLNTRG,SHA,SHB,SGA,SGOTP,KGRDT,        GSTHS124
     1 SRGOTP,CTP,SCALR,SCALT,GUSTV,RHOR)                                GSTHS125
C                                                                        GSTHS126
C                                                                        GSTHS127
      IF(IND.NE.0)GO TO 95                                               GSTHS128
      IF(GUSTV.GT.0)GO TO 95                                             GSTHS129
      IF(GUSTV.LE.VELGL)GO TO 95                                         GSTHS130
      DELTG=10.*DELTG                                                    GSTHS131
      IND=1                                                              GSTHS132
   95 CONTINUE                                                           GSTHS133
C                                                                        GSTHS134
      VELG(I)=GUSTV                                                      GSTHS135
      RHO(I)=RHOR                                                        GSTHS136
C                                                                        GSTHS137
      IF(VELG(I).GT.-2.0.AND.IND.GT.0.AND.VELG(I).GT.VELGL)GO TO 101     GSTHS138
      VELGL=VELG(I)                                                      GSTHS139
      IF(KPRTMH.EQ.0)GO TO 98                                            GSTHS140
      LINES=LINES+1                                                      GSTHS141
      IF(LINES.LE.KREW)GO TO 97                                          GSTHS142
      CALL HEADNG                                                        GSTHS143
      WRITE(NOUT,2)                                                      GSTHS144
   97 WRITE(NOUT,1)I,TIMG(I),VELG(I),RHO(I),SLNTRG                       GSTHS145
   98 CONTINUE                                                           GSTHS146
      T=T+DELTG                                                          GSTHS147
  100 CONTINUE                                                           GSTHS148
  101 CONTINUE                                                           GSTHS149
      NTMGST=ICNT+1                                                      GSTHS150
      DT1=TIMG(ICNT)-TIMG(ICNT-1)                                        GSTHS151
      DV1=VELG(ICNT)-VELG(ICNT-1)                                        GSTHS152
      TIMG(NTMGST)=TIMG(ICNT-1)-DT1*VELG(ICNT-1)/DV1                     GSTHS153
      VELG(NTMGST)=0.                                                    GSTHS154
      RHO(NTMGST)=RHCA                                                   GSTHS155
      IF(KPRTMH.EQ.0)GO TO 102                                           GSTHS156
      WRITE(NOUT,1)ICNT,TIMG(ICNT),VELG(ICNT),RHO(ICNT),SLNTRG           GSTHS157
```

```
      WRITE(NOUT,1)NTMGST,TIMG(NTMGST),VELG(NTMGST),RHO(NTMGST)           GSTHS158
                                                                          GSTHS159
1     FORMAT(1H ,I3,2(1X,F8.4),1X,E12.5,1X,F12.5)                         GSTHS160
                                                                          GSTHS161
102   CONTINUE                                                            GSTHS162
                                                                          GSTHS163
      VGUST=VELG(1)                                                       GSTHS164
                                                                          GSTHS165
      RETURN                                                              GSTHS166
      END                                                                 GSTHS167
```

```
      SUBROUTI[illegible] (TMTA,Z,C2,GUSTR,RHOR)                           GUSRHO 2
C     GIVEN R, [illegible] C2, SUBROUTINE GUSRHO RETURNS GUSTR AND RHOR.GUSRHO 3
C     R IS THE [illegible].                                               GUSRHO 4
C     TMTA = TI[illegible] ARRIVAL TIME.                                  GUSRHO 5
C     Z IS THE [illegible] PRESSURE RATIO.                                GUSRHO 6
C     C2 IS THE [illegible] ING FACTOR.                                   GUSRHO 7
C     GUSTR IS [illegible] OF THE GUST VELOCITY (MATERIAL VELOCITY) TO    GUSRHO 8
C     THE AMBIEN[illegible] OF SOUND.                                     GUSRHO 9
C     RHOR IS THE [illegible] OF THE AIR OVERDENSITY TO THE AMBIENT AIR DENSITGUSRHO10
      DIMENSI[illegible]), AP(7), BP(7), CP(7), RR(7), AR(7), BR(7), CR(7), GUSRHO11
     1RG(5), AG(5), BG(5), CG(5)                                          GUSRHO12
      DATA RP/3.8,1.7,0.6,0.45,0.35,0.3,0.115/,AP/.272444,.171847,.02174GUSRHO13
     19,.145,.0724,.1662,.167066/,BP/.029527,.080934,.277338,-.220833,.1GUSRHO14
     237,-.11,.0804?8/,CP/-.001097,-.007659,-.071254,.416667,-.02,-.08,-GUSRHO15
     33.724616/,RR/4.2,1.5,1.12,0.85,0.5,0.28,0.115/,AR/.854207,1.1148,2.GUSRHO16
     4435118,-1.80837,-.345,-1.637971,.0146514/,BR/.0224069,-.0654667,-1.GUSRHO17
     5580976,6.01944,3.03571,7.98454,-.003846/,CR/-.00108276,.00506667,.6GUSRHO18
     618421,-3.05555,-1.57143,-6.2972,1.153846/,RG/2.5,1.0,0.5,0.335,0.1GUSRHO19
     7715/,AG/1.432033,1.944445,3.22,3.928788,2.0443/,BG/-.114293,-.54222GUSRHO20
     883,-3.16,-3.554545,1.813868/,CG/.008592,.097778,1.44,-.606061,.1608GUSRHO21
     975/                                                                 GUSRHO22
      IF (TMTA.GE.0.0) GO TO 10                                           GUSRHO23
      GUSTR=0.0                                                           GUSRHO24
      RHOR=0.0                                                            GUSRHO25
      RETURN                                                              GUSRHO26
10    IF (R.LT.10.0) GO TO 20                                             GUSRHO27
      TOP=C2*(.252609+.089206*ALOG(R))                                    GUSRHO28
      GO TO 70                                                            GUSRHO29
20    DO 30 I=1,7                                                         GUSRHO30
      IF (R.GE.RP(I)) GO TO 40                                            GUSRHO31
30    CONTINUE                                                            GUSRHO32
      WRITE (6,230) R                                                     GUSRHO33
      STOP                                                                GUSRHO34
40    TOP=C2*((CP(I)*R+BP(I))*R+AP(I))                                    GUSRHO35
      DO 50 I=1,7                                                         GUSRHO36
      IF (R.GE.RR(I)) GO TO 60                                            GUSRHO37
50    CONTINUE                                                            GUSRHO38
60    TORHO=TOP*((CR(I)*R+BR(I))*R+AR(I))                                 GUSRHO39
      GO TO 90                                                            GUSRHO40
70    TORHO=TOP*(1.0-3.0/R**2)                                            GUSRHO41
      IF (R.LT.17.0) GO TO 80                                             GUSRHO42
      TOGUST=TOP*(1.0+76.206/R**3)                                        GUSRHO43
      GO TO 130                                                           GUSRHO44
80    TOGUST=TOP*(-.0000083*R**3+1.056289)                                GUSRHO45
      GO TO 130                                                           GUSRHO46
90    IF (R.LT.7.0) GO TO 100                                             GUSRHO47
      TOGUST=TOP*(-.001667*R+1.064659)                                    GUSRHO48
      GO TO 130                                                           GUSRHO49
100   DO 110 I=1,5                                                        GUSRHO50
      IF (R.GE.RG(I)) GO TO 120                                           GUSRHO51
110   CONTINUE                                                            GUSRHO52
120   TOGUST=TOP*((CG(I)*R+BG(I))*R+AG(I))                                GUSRHO53
```

```
130   IF (Z.GE.11.057506) GO TO 140                                            GUSRH054
      IF (Z.GT.2.3609977) GO TO 150                                            GUSRH055
      ASP=0.5                                                                  GUSRH056
      GO TO 160                                                                GUSRH057
140   ASP=0.0                                                                  GUSRH058
      GO TO 160                                                                GUSRH059
150   ASP=(-.0332*Z+.388)*Z-.231                                               GUSRH060
160   IF (Z.GE.9.5180723) GO TO 170                                            GUSRH061
      IF (Z.GT.5.0/3.0) GO TO 180                                              GUSRH062
      BSP=1.0                                                                  GUSRH063
      IF (Z.GT.1.5589942) GO TO 190                                            GUSRH064
      GO TO 200                                                                GUSRH065
170   BSP=1.67-.011*Z                                                          GUSRH066
      GO TO 190                                                                GUSRH067
180   BSP=.88+.072*Z                                                           GUSRH068
190   IF (TMTA.GE.TOGUST.AND.TMTA.GE.TORHO) GO TO 200                          GUSRH069
      CSP=8.71+.1843*Z-104.0/(Z+10.0)                                          GUSRH070
      CSPG=0.0                                                                 GUSRH071
      IF (TMTA.LT.TOGUST) CSPG=CSP                                             GUSRH072
      CSPR=0.0                                                                 GUSRH073
      IF (TMTA.LT.TORHO) CSPR=CSP                                              GUSRH074
      GO TO 210                                                                GUSRH075
200   AGUST=ASP+BSP*Z                                                          GUSRH076
      ARHO=AGUST                                                               GUSRH077
      GO TO 220                                                                GUSRH078
210   AGUST=ASP+BSP*Z/(1.0+CSPG*TMTA/TOGUST)                                   GUSRH079
      ARHO=ASP+BSP*Z/(1.0+CSPR*TMTA/TORHO)                                     GUSRH080
220   GUSTR=(5.0*Z/SQRT(7.0*(7.0+6.0*Z)))*(1.0-TMTA/TOGUST)*EXP(-AGUST*TGUSRH081
     1MTA/TOGUST)                                                              GUSRH082
      RHOR=(5.0*Z/(7.0+Z))*(1.0-TMTA/TORHO)*EXP(-ARHO*TMTA/TORHO)              GUSRH083
      RETURN                                                                   GUSRH084
C                                                                              GUSRH085
C                                                                              GUSRH086
230   FORMAT (17H SCALED RANGE OF ,E11.4,55H KILOFEET IS LESS THAN MINIMGUSRH087
     1UM PERMISSIBLE VALUE, 0.115)                                             GUSRH088
      END                                                                      GUSRH089
```

```
      SUBROUTINE GUSTDR(CR,NTGST,PSTR,PATR,NOR,REST,                    GUSTDR 2
     2 VELG,TIMG,RHO,OMEGA,PLOTF,PLOTA,                                 GUSTDR 3
     2 AS,AA,AST,AAT,                                                   GUSTDR 4
     3 PS,PA,PST,PAT,TIME,PKL,STALDS,NTMRSP,                            GUSTDR 5
     4 STRESS,STR,STL,ALLOWS,SYMCOD,ASMCOD,NINTLD,FTG,GEOMBX,GEOMBD)    GUSTDR 6
C                                                                       GUSTDR 7
      COMMON/ZZZ/CASE(48),NIN,NOUT,KROW,LINES,IPRNT,NER                 GUSTDR 8
      COMMON/DISK2/ND2,ITBL2(843),NRECSA,IBUMP,NKD,VOBWS(20)            DISK2  2
C                                                                       GUSTDR10
      COMMON /XTF/NF(100)                                               GUSTDR11
      EQUIVALENCE (NF(9),NFREQ)                                         GUSTDR12
     1 , (NF(2),GAMX)                                                   GUSTDR13
     2 , (NF(43),NBCX),(NF(44),NSBETO)                                  GUSTDR14
     3 , (NF(1),AMACH),(NF(3),VSS)                                      GUSTDR15
     4 , (NF(100),SIZFCT)                                               GUSTDR16
      EQUIVALENCE (NF(13),RHOO),      (NF(14),SIGMA)                    GUSTDR17
     1        ,   (NF(18),VEL),       (NF(19),ALT)                      GUSTDR18
     1 , (NF(29),NACC)                                                  GUSTDR19
     1 , (NF(33),NSTRSS)                                                GUSTDR20
     3 , (NF(23),KPRTRM)                                                GUSTDR21
     1 , (NF(54),KMAN),(NF(55),AB),(NF(56),AC),(NF(57),INDSYM)          GUSTDR22
     4 , (NF(51),AN)                                                    GUSTDR23
      EQUIVALENCE (NF(61),NORMAX),  (NF(62),TIMEMX),   (NF(63),EER)     GUSTDR24
     1 , (NF(64),KGRD),(NF(65),KLPT),(NF(66),HGRD)                      GUSTDR25
     2 , (NF(67),KLOAD),(NF(68),NCRITS)                                 GUSTDR26
     3 , (NF(89),TAU)                                                   GUSTDR27
     4 , (NF(97),DELT)                                                  GUSTDR28
     5 , (NF(4),PO)                                                     GUSTDR29
C                                                                       GUSTDR30
      DIMENSION PSTR(1),PATR(1),NOR(1),REST(1),VELG(1),TIMG(1)          GUSTDR31
      DIMENSION RHO(1),OMEGA(1),PLOTF(1),PLOTA(1)                       GUSTDR32
      DIMENSION AS(1), AA(1), AST(1), AAT(1)                            GUSTDR33
      DIMENSION PS(1), PA(1)                                            GUSTDR34
      DIMENSION PST(1), PAT(1), TIME(1)                                 GUSTDR35
      DIMENSION PKL(1),STRESS(1),STR(1),STL(1)                          GUSTDR36
      DIMENSION ALLOWS(NINTLD,1),STALDS(NINTLD,1)                       GUSTDR37
      DIMENSION SYMCOD(1),ASMCOD(1)                                     GUSTDR38
      DIMENSION CR(3,1)                                                 GUSTDR39
      DIMENSION FTG(2,1)                                                GUSTDR40
      DIMENSION GEOMBX(1),GEOMBD(1)                                     GUSTDR41
      DIMENSION IGTABL(20)                                              GUSTDR42
C                                                                       GUSTDR43
      DATA ITERMX/5/                                                    GUSTDR44
      DATA EPTHTA/1.0E-02/                                              GUSTDR45
C     TABLE FOR A.S. EQUIVALENCE OF AERO ORIENT TO BLST ORIENT          GUSTDR46
      DATA IGTABL/1,2,3,4,5,6,7,8,9,10,11,-11,-10,-9,12,-12,-13,13,0,0/ GUSTDR47
C                                                                       GUSTDR48
C                                                                       GUSTDR49
    1 FORMAT (6I12)                                                     GUSTDR50
    2 FORMAT (6F12.0)                                                   GUSTDR51
C                                                                       GUSTDR52
C                                                                       GUSTDR53
```

```
C         REVISE MAX AND MIN ALLOW LOADS IF RQD                           GUSTDR54
C                                                                         GUSTDR55
          IF (KLOAD.EQ.0) GO TO 60                                        GUSTDR56
C                                                                         GUSTDR57
          DO 50 L=1,NINTLD                                                GUSTDR58
   50     READ(NIN,2)(STALDS(L,J),J=7,8)                                  GUSTDR59
C                                                                         GUSTDR60
   60     CONTINUE                                                        GUSTDR61
C                                                                         GUSTDR62
C                                                                         GUSTDR63
C         NOW HALVE THOSE C/L ALLOWABLES                                  GUSTDR64
C                                                                         GUSTDR65
          DO 61 L=1,NINTLD                                                GUSTDR66
          IF(SYMCOD(L).NE.0.AND.ASMCOD(L).NE.0)GO TO 61                   GUSTDR67
          STALDS(L,7)=0.50*STALDS(L,7)                                    GUSTDR68
          STALDS(L,8)=0.50*STALDS(L,8)                                    GUSTDR69
   61     CONTINUE                                                        GUSTDR70
C                                                                         GUSTDR71
C                                                                         GUSTDR72
C         INPUT MAX AND MIN STRESSES IF RQD                               GUSTDR73
C                                                                         GUSTDR74
          IF(NCRITS.EQ.0)GO TO 70                                         GUSTDR75
C                                                                         GUSTDR76
          DO 65 L=1,NSTRSS                                                GUSTDR77
          READ(NIN,2)(ALLOWS(L,J),J=1,2)                                  GUSTDR78
          IF(SYMCOD(L).NE.0.AND.ASMCOD(L).NE.0)GO TO 65                   GUSTDR79
          ALLOWS(L,1)=0.50*ALLOWS(L,1)                                    GUSTDR80
          ALLOWS(L,2)=0.50*ALLOWS(L,2)                                    GUSTDR81
   65     CONTINUE                                                        GUSTDR82
C                                                                         GUSTDR83
   70     CONTINUE                                                        GUSTDR84
C                                                                         GUSTDR85
C         NOW GET SOLN TIMES                                              GUSTDR86
          IND=0                                                           GUSTDR87
          DELTT=DELT                                                      GUSTDR88
          TI=-DELT                                                        GUSTDR89
          DO 100 I=1,NTMRSP                                               GUSTDR90
          TI=TI+DELTT                                                     GUSTDR91
          TIME(I)=TI                                                      GUSTDR92
          IF(IND.GT.1)GO TO 100                                           GUSTDR93
          IF(TI.LT.0.25)GO TO 100                                         GUSTDR94
          IF(TI.LE.1.0)GO TO 99                                           GUSTDR95
          IND=2                                                           GUSTDR96
          TI=TI-DELTT                                                     GUSTDR97
          DELTT=DELTT*2.0                                                 GUSTDR98
          TI=TI+DELTT                                                     GUSTDR99
          TIME(I)=TI                                                      GUSTD100
          GO TO 100                                                       GUSTD101
   99     IF(IND.EQ.1)GO TO 100                                           GUSTD102
          IND=1                                                           GUSTD103
          TI=TI-DELTT                                                     GUSTD104
          DELTT=DELTT*5.0                                                 GUSTD105
```

```
      TI=TI+DELTT                                                       GUSTD106
      TIME(I)=TI                                                        GUSTD107
  100 CONTINUE                                                          GUSTD108
C                                                                       GUSTD109
      IF(TIME(NTMRSP).GT.TIMEMX)TIME(NTMRSP)=TIMEMX                     GUSTD110
C                                                                       GUSTD111
C     LOOP THRU DESIRED ORIENTATIONS                                    GUSTD112
C                                                                       GUSTD113
      DO 2000 LOOP=1,NORMAX                                             GUSTD114
      CALL HEADNG                                                       GUSTD115
      NG=NOR(LOOP)                                                      GUSTD116
      SLNTRG=REST(LOOP)                                                 GUSTD117
C                                                                       GUSTD118
C     NG IS BASE ORIENTATION                                            GUSTD119
C                                                                       GUSTD120
C     CALC INDSYM FOR GUST                                              GUSTD121
C                                                                       GUSTD122
C     INDSYM=1 IS A SYMMETRIC CASE                                      GUSTD123
C     INDSYM=2 IS AN UNSYMMETRIC CASE                                   GUSTD124
C                                                                       GUSTD125
      NGL=IGTABL(NG)                                                    GUSTD126
C     NGL IS EQUIV. NG FOR A.S. CASES                                   GUSTD127
C                                                                       GUSTD128
      NGLL=IABS(NGL)                                                    GUSTD129
C                                                                       GUSTD130
      OMEGMX=OMEGA(NFREQ)                                               GUSTD131
      GAMX=CR(1,NGLL)                                                   GUSTD132
      GAMY=CR(2,NGLL)                                                   GUSTD133
      GAMZ=CR(3,NGLL)                                                   GUSTD134
      IF(NGL.LT.0)GAMY=-GAMY                                            GUSTD135
      INDSYM=1                                                          GUSTD136
      IF(GAMY.NE.0)INDSYM=2                                             GUSTD137
C                                                                       GUSTD138
      LOC=NKD*IBUMP+3                                                   GUSTD139
      NW=6*NBOX                                                         GUSTD140
      CALL READMS(ND2,GEOMBX,NW,LOC)                                    GUSTD141
      LOC=LOC+1                                                         GUSTD142
      NW=6*NSBFTO                                                       GUSTD143
      CALL READMS(ND2,GEOMBD,NW,LOC)                                    GUSTD144
C                                                                       GUSTD145
C     NOW FIND GEOMETRIC LOC OF BLAST  INTERCEPTION ON A/C              GUSTD146
C                                                                       GUSTD147
      ALI=1.E+10                                                        GUSTD148
C                                                                       GUSTD149
      DO 30 I1=1,NBOX                                                   GUSTD150
      I2=I1+NBOX                                                        GUSTD151
      I3=I2+NBOX                                                        GUSTD152
      I4=I3+NBOX                                                        GUSTD153
      I5=I4+NBOX                                                        GUSTD154
      I6=I5+NBOX                                                        GUSTD155
      XINT=(GEOMBX(I1)+GEOMBX(I4))/2.0                                  GUSTD156
      YINT=(GEOMBX(I2)+GEOMBX(I5))/2.0                                  GUSTD157
```

```
      ZINT=(GEOMBX(I3)+GEOMBX(I6))/2.0                                  GUSTD158
      YINTL=-YINT                                                       GUSTD159
      DY=GEOMBX(I5)-GEOMBX(I2)                                          GUSTD160
      DZ=GEOMBX(I6)-GEOMBX(I3)                                          GUSTD161
      DDL=SQRT(DY**2+DZ**2)                                             GUSTD162
      SINDIH=DZ/DDL                                                     GUSTD163
      COSDIH=DY/DDL                                                     GUSTD164
      THETAR=GAMZ*COSDIH-GAMY*SINDIH                                    GUSTD165
      THETAL=GAMZ*COSDIH+GAMY*SINDIH                                    GUSTD166
      IF(ABS(THETAR).LT.EPTHTA)GO TO 25                                 GUSTD167
      ALR=GAMX*XINT+GAMY*YINT+GAMZ*ZINT                                 GUSTD168
      IF(ALR.GT.ALI)GO TO 25                                            GUSTD169
      ALI=ALR                                                           GUSTD170
      X=XINT                                                            GUSTD171
      Y=YINT                                                            GUSTD172
      Z=ZINT                                                            GUSTD173
   25 CONTINUE                                                          GUSTD174
      IF(ABS(THETAL).LT.EPTHTA)GO TO 30                                 GUSTD175
      ALL=GAMX*XINT+GAMY*YINTL+GAMZ*ZINT                                GUSTD176
      IF(ALL.GT.ALI)GO TO 30                                            GUSTD177
      ALI=ALL                                                           GUSTD178
      X=XINT                                                            GUSTD179
      Y=YINTL                                                           GUSTD180
      Z=ZINT                                                            GUSTD181
   30 CONTINUE                                                          GUSTD182
C                                                                       GUSTD183
      DO 40 I1=1,NSBETO                                                 GUSTD184
      I2=I1+NSBETO                                                      GUSTD185
      I3=I2+NSBETO                                                      GUSTD186
      I4=I3+NSBETO                                                      GUSTD187
      I5=I4+NSBETO                                                      GUSTD188
      I6=I5+NSBETO                                                      GUSTD189
      XINT=(GEOMBD(I1)+GEOMBD(I4))/2.0                                  GUSTD190
      YINT=(GEOMBD(I2)+GEOMBD(I5))/2.0                                  GUSTD191
      ZINT=(GEOMBD(I3)+GEOMBD(I6))/2.0                                  GUSTD192
      YINTL=-YINT                                                       GUSTD193
      ALR=GAMX*XINT+GAMY*YINT+GAMZ*ZINT                                 GUSTD194
      IF(ALR.GT.ALI)GO TO 35                                            GUSTD195
      ALI=ALR                                                           GUSTD196
      X=XINT                                                            GUSTD197
      Y=YINT                                                            GUSTD198
      Z=ZINT                                                            GUSTD199
   35 CONTINUE                                                          GUSTD200
      ALI=GAMX*XINT+GAMY*YINTL+GAMZ*ZINT                                GUSTD201
      IF(ALL.GT.ALI)GO TO 40                                            GUSTD202
      ALI=ALL                                                           GUSTD203
      X=XINT                                                            GUSTD204
      Y=YINTL                                                           GUSTD205
      Z=ZINT                                                            GUSTD206
   40 CONTINUE                                                          GUSTD207
C                                                                       GUSTD208
C     NOW THE TIME OF BURST INTERCEPT (WRT ORIGIN OF AAS) IS            GUSTD209
```

```
      TAU=(ALI*SIZFCT/12.)/(VSS*(1.0+GAMX*AMACH))                        GUSTD210
      IF(TAU.GT.0)TAU=0.                                                  GUSTD211
C                                                                         GUSTD212
      WRITE(NOUT,3)KGRD,KLPT,HGRD,NG,EFR,SLNTRG,TIMEMX,NTMRSP,            GUSTD213
     1 GAMX,GAMY,GAMZ,TAU,X,Y,Z                                           GUSTD214
C                                                                         GUSTD215
    3 FORMAT(1HO,20X, BLAST TIME RESPONSE PARAMETERS /1HO,                GUSTD216
     1 20X, KGRD =    ,I4/1H ,                                            GUSTD217
     1 20X, KLPT =    ,I4/1H ,                                            GUSTD218
     1 20X, HGRD = ,F8.2,  FT. /1H ,                                      GUSTD219
     1 20X, ORIENTATION NO. =   ,I4/1H ,                                  GUSTD220
     1 20X, YIELD             =   ,F10.4,  KILOTONS /1H ,                 GUSTD221
     1 20X, SLNTRG            =   ,F15.4,  FEET /1H ,                     GUSTD222
     1 20X, MAX. RESP. TIME =   ,F10.4,  SECS /1H ,                       GUSTD223
     1 20X, NO. TIME RSP.PTS=   ,I5/1H ,                                  GUSTD224
     1 20X, GAMX =   ,E12.5/1H ,                                          GUSTD225
     1 20X, GAMY =   ,E12.5/1H ,                                          GUSTD226
     1 20X, GAMZ =   ,E12.5/1H ,                                          GUSTD227
     1 20X, TAU =  ,F10.4,  SECS /1H ,                                    GUSTD228
     1 20X, XINT =   ,F10.2/1H ,                                          GUSTD229
     1 20X, YINT =   ,F10.2/1H ,                                          GUSTD230
     1 20X, ZINT =   ,F10.2)                                              GUSTD231
C                                                                         GUSTD232
C     NEW LOOP FOR CRITICAL RANGE IF ROD                                  GUSTD233
C                                                                         GUSTD234
      ITER=1                                                              GUSTD235
  500 CONTINUE                                                            GUSTD236
C                                                                         GUSTD237
      IF(ITER.GT.1)CALL HEADNG                                            GUSTD238
      WRITE(NOUT,5)ITER                                                   GUSTD239
    5 FORMAT(1HO,20X, RANGE ITERATION NO. ,I4)                            GUSTD240
C                                                                         GUSTD241
      IF(ITER.GT.ITERMX)GO TO 1000                                        GUSTD242
C                                                                         GUSTD243
      CALL FLTPOS(0.,SLNTRG,VEL,ALT,AN,AB,AC,GAMX,GAMY,GAMZ,KMAN,         GUSTD244
     1 DUM,XB,YB,ZB,1,XE,YE,ZE)                                           GUSTD245
C                                                                         GUSTD246
C                                                                         GUSTD247
      NTMGST=NTGST-1                                                      GUSTD248
      DELTG=0.050                                                         GUSTD249
C                                                                         GUSTD250
      CALL GSTHST(                                                        GUSTD251
     1 RHOA,NTMGST,DELTG,SCALT,VELG,TIMG,RHO,VSS,DELPR,VGUST,SCALR,PO,    GUSTD252
     1 EFR,VEL,SLNTRG,ALT,TARR,AN,AB,AC,GAMX,GAMY,GAMZ,KMAN,              GUSTD253
     1 KGRD,HGRD,XB,YB,ZB,XE,YE,ZE)                                       GUSTD254
C                                                                         GUSTD255
      CALL TIMHST(FTG,                                                    GUSTD256
     1 NINTLD,NFREQ,NTMRSP,NTMGST,TIMEMX,OMEGMX,INDSYM,RHOA,              GUSTD257
     2 VELG,TIMG,RHO,OMEGA,PLOTF,PLOTA,                                   GUSTD258
     3 AS,AA,AST,AAT,NACC,NG,NGL,TIMG,                                    GUSTD259
     4 PS,PA,PST,PAT,TIMF)                                                GUSTD260
C                                                                         GUSTD261
```

```
      CALL LOADCH(TIMG,ITERS,
     1 NINTLD,NTMRSP,NG,NGL,GAMX,GAMY,GAMZ,INDSYM,PKL,PST,PAT,
     1 PSTR,PATR,STALDS,TIME,VGUST,VGUSTR,PO,DELPR,DELPRR,VSS)
C
      IF(NSTRSS.NE.0)CALL STRSCH(
     1 ITERS,
     1 NINTLD,NTMRSP,PKL,PST,PAT,STL,STR,ALLOWS,TIME,VGUST,VGUSTR,PO,
     1 DELPR,DELPRR,VSS,STRESS,NSTRSS,NCRITS)
C
C
C     FIND NEW RANGE FOR ITERATION
C
      CALL RANGE(DELPRR,RTA)
      RR=1.0
      IF(RTA.GT.0.55)RR=1.0+0.06*(RTA-0.55)**(1./3.)
      RTA=RTA/RR
      SLNTRG=RTA/SCALR
      TARR=0.
      CALL TAR(RTA,TARR)
      TARR=-TARR*SCALT
      CALL FLTPCS(TARR,SLNTRG,VEL,ALT,AN,AB,AC,GAMX,GAMY,GAMZ,KMAN,
     1 DUM,XB,YB,ZB,1,XE,YE,ZE)
      IF(KIPT.EQ.0)GO TO 1000
      IF(ABS(1.0-VGUST/VGUSTR).LT.0.025)ITERS=1
      IF(ITERS.NE.0)GO TO 1000
C
      ITER=ITER+1
      GO TO 500
C
 1000 CONTINUE
C
      IF(ITERS.NE.0)WRITE(NOUT,4)ITER,SLNTRG
    4 FORMAT(1H0,20X, CONVERGED SOLUTION /1H ,
     1 20X,I2,    ITERATIONS /1H ,
     1 20X, CRITICAL RANGE IS   ,F10.2, FT. )
C
      CALL HEADNG
      WRITE(NOUT,8)
      DO 700 I=1,NINTLD
      LINES=LINES+1
      IF(LINES.LT.KROW)GO TO 690
      CALL HEADNG
      WRITE(NOUT,8)
  690 CONTINUE
  700 WRITE(NOUT,9)I,(STALDS(I,J),J=1,6)
    9 FORMAT(1H ,I4,3(1X,F4.1),3(1X,E12.5))
    8 FORMAT(1H0,20X, INTEGRATED LOAD DEFINITIONS /1H0,
     1  LOAD BEAM CODE COMP ,9X, XAAS ,9X, YAAS ,9X, ZAAS )
C
C
 2000 CONTINUE
C
```

```
      RETURN                                                            GLSTD314
      END                                                               GUSTD315
```

```
      SUBROUTINE IFT(
     1 NFREQ,OMEGA,OMEGMX,FACTO,NFACT,NMP,A,NTP,TIME,G,TIMEMX)
C
      COMMON/XIF/NF(100)
      EQUIVALENCE (NF(91),FINFRQ)
C
      DIMENSION OMEGA(1)
      DIMENSION TIME(1)
      DIMENSION A(2,NMP,1)
      DIMENSION G(NMP,1)
C
      DATA TWOOPI/0.636619772/
C
      DO 70 K=1,NTP
      DO 60 J=1,NMP
   60 G(J,K)=0.0
   70 CONTINUE
C
C     INT. FROM 0 TO FIRST FREQ IF FIRST FREQ NOT 0
C
      DEL = 1.0/(OMEGA(2)-OMEGA(1))
      DO 80 K=2,NTP
      T=TIME(K)
      DEN=1.0/T
      ANG=OMEGA(1)*T
      C2=COS(ANG)
      S2=SIN(ANG)
      DO 80 J=1,NMP
      AI1 = DEL*(A(2,J,1)*OMEGA(2)-A( 2,J, 2)*OMEGA( 1))
      AI2 = A(2,J,1)
      BBI = DEL*(A(2,J,2)-A(2,J,1))
      G(J,K) = G(J,K) + TWOOPI*DEN*(AI2*C2-AI1-BBI*DEN*S2)
   80 CONTINUE
C
   90 CONTINUE
C
C
      DO 120 K=2,NTP
      T=TIME(K)
      DEN=1.0/T
      ANG1=OMEGA(1)*T
      C1=COS(ANG1)
      S1=SIN(ANG1)
      DO 110 I=2,NFREQ
      DEL=1.0/(OMEGA(I)-OMEGA(I-1))
      ANG2=OMEGA(I)*T
      C2=COS(ANG2)
      S2=SIN(ANG2)
      DO 100 J=1,NMP
C
      AI2 = A(2,J,I)
      AI1 = A(2,J,I-1)
```

```
      BBI = DEL*(AI2-AI1)
      G(J,K) = G(J,K) + TWOOPI*DEN*(AI2*C2-AI1*C1-BBI*DEN*(S2-S1))
  100 CONTINUE
      C1=C2
      S1=S2
  110 CONTINUE
  120 CONTINUE
C
C     NOW FINISH TAIL INTEGRATION FOR FREQS ABOVE LAST INPUT FREQ
C
      DEL=1.0/(FINFRQ-OMEGA(NFREQ))
      DO 150 K=2,NTP
      T=TIME(K)
      DEN=1.0/T
      ANG1=OMEGA(NFREQ)*T
      C1=COS(ANG1)
      S1=SIN(ANG1)
      ANG2=FINFRQ*T
      C2=COS(ANG2)
      S2=SIN(ANG2)
      DO 150 J=1,NMP
      AI1 = A(2,J,NFREQ)
      BBI = -A(2,J,NFREQ)*DEL
      G(J,K) = G(J,K) + TWOOPI*DEN*(-AI1*C1-BBI*DEN*(S2-S1))
  150 CONTINUE
C
      RETURN
      END
```

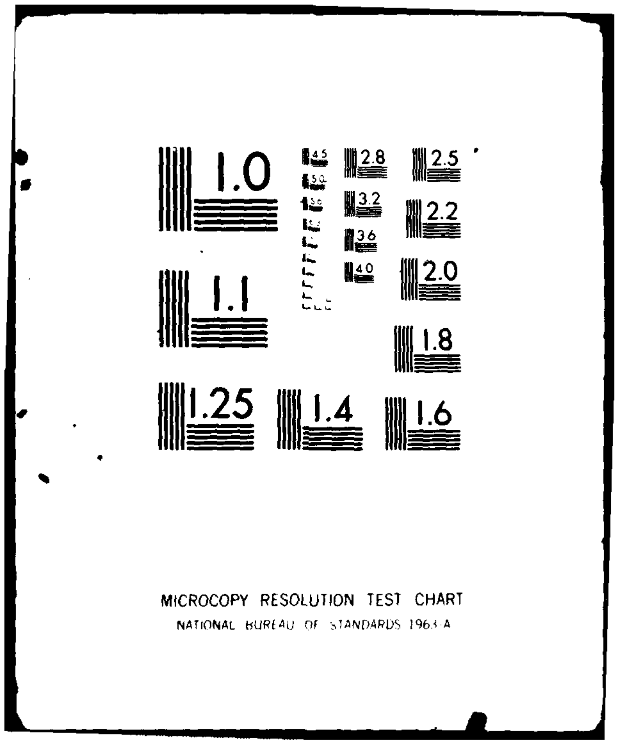
```
      SUBROUTINE INTRPB(X,Y,Z,NX,NY,NZ,XT,YT,ZT,F,FF)                   INTRPB 2
C     SUBROUTINE INTERP PERFORMS LINEAR INTERPOLATION OF TABULATED      INTRPB 3
C     FUNCTIONS OF THREE OR LESS VARIABLES.                             INTRPB 4
C     FOR EACH VALUE OF X, FUNCTION MUST BE TABULATED FOR SAME          INTRPB 5
C     VALUES OF Y AND Z.                                                INTRPB 6
C     X=VALUE OF FIRST ARGUMENT                                         INTRPB 7
C     Y=VALUE OF SECOND ARGUMENT                                        INTRPB 8
C     Z=VALUE OF THIRD ARGUMENT                                         INTRPB 9
C     XT=TABLE VALUES OF X                                              INTRPB10
C     YT=TABLE VALUES OF Y                                              INTRPB11
C     ZT=TABLE VALUES OF Z                                              INTRPB12
C     NX=ORDER OF XT                                                    INTRPB13
C     NY=ORDER OF YT (0 IF SINGLE INTERPOLATION)                        INTRPB14
C     NZ=ORDER OF ZT (0 IF SINGLE OR DOUBLE INTERPOLATION)              INTRPB15
C     F=TABLE OF FUNCTION, STORED WITH                                  INTRPB16
C       THIRD ARGUMENT VARYING MOST RAPIDLY, ETC                        INTRPB17
C     FF=STORAGE FOR INTERPOLATED VALUE                                 INTRPB18
      DIMENSION F(1), XT(1), YT(1), ZT(1)                               INTRPB19
      IF (X.LT.XT(1)) GO TO 110                                         INTRPB20
      DO 10 I=2,NX                                                      INTRPB21
      IF (X.LE.XT(I)) GO TO 20                                          INTRPB22
   10 CONTINUE                                                          INTRPB23
      GO TO 110                                                         INTRPB24
   20 IX=I-1                                                            INTRPB25
      IF (NY.EQ.0) GO TO 90                                             INTRPB26
      IF (Y.LT.YT(1)) GO TO 120                                         INTRPB27
      DO 30 I=2,NY                                                      INTRPB28
      IF (Y.LE.YT(I)) GO TO 40                                          INTRPB29
   30 CONTINUE                                                          INTRPB30
      GO TO 120                                                         INTRPB31
   40 IY=I-1                                                            INTRPB32
      IF (NZ.EQ.0) GO TO 70                                             INTRPB33
      IF (Z.LT.ZT(1)) GO TO 130                                         INTRPB34
      DO 50 I=2,NZ                                                      INTRPB35
      IF (Z.LE.ZT(I)) GO TO 60                                          INTRPB36
   50 CONTINUE                                                          INTRPB37
      GO TO 130                                                         INTRPB38
   60 IZ=I-1                                                            INTRPB39
      I111=(IX-1)*NY*NZ+(IY-1)*NZ+IZ                                    INTRPB40
      I112=I111+1                                                       INTRPB41
      I121=I111+NZ                                                      INTRPB42
      I122=I121+1                                                       INTRPB43
      I211=I111+NY*NZ                                                   INTRPB44
      I212=I211+1                                                       INTRPB45
      I221=I211+NZ                                                      INTRPB46
      I222=I221+1                                                       INTRPB47
      DZ=ZT(IZ+1)-ZT(IZ)                                                INTRPB48
      DZ1=(ZT(IZ)-Z)/DZ                                                 INTRPB49
      DZ2=(ZT(IZ+1)-Z)/DZ                                               INTRPB50
      F11=DZ2*F(I111)-DZ1*F(I112)                                       INTRPB51
      F12=DZ2*F(I121)-DZ1*F(I122)                                       INTRPB52
      F21=DZ2*F(I211)-DZ1*F(I212)                                       INTRPB53
```

```
      F22=DZ2*F(I221)-DZ1*F(I222)                                       INTRPB54
      GO TO 80                                                          INTRPB55
70    I11=(IX-1)*NY+IY                                                  INTRPB56
      I12=I11+1                                                         INTRPB57
      I21=I11+NY                                                        INTRPB58
      I22=I21+1                                                         INTRPB59
      F11=F(I11)                                                        INTRPB60
      F12=F(I12)                                                        INTRPB61
      F21=F(I21)                                                        INTRPB62
      F22=F(I22)                                                        INTRPB63
80    DY=YT(IY+1)-YT(IY)                                                INTRPB64
      DY1=(YT(IY)-Y)/DY                                                 INTRPB65
      DY2=(YT(IY+1)-Y)/DY                                               INTRPB66
      F1=DY2*F11-DY1*F12                                                INTRPB67
      F2=DY2*F21-DY1*F22                                                INTRPB68
      GO TO 100                                                         INTRPB69
90    F1=F(IX)                                                          INTRPB70
      F2=F(IX+1)                                                        INTRPB71
100   DX=XT(IX+1)-XT(IX)                                                INTRPB72
      DX1=(XT(IX)-X)/DX                                                 INTRPB73
      DX2=(XT(IX+1)-X)/DX                                               INTRPB74
      FF=DX2*F1-DX1*F2                                                  INTRPB75
      RETURN                                                            INTRPB76
110   WRITE (6,150) X,(XT(I),I=1,NX)                                    INTRPB77
      GO TO 140                                                         INTRPB78
120   WRITE (6,160) Y,(YT(I),I=1,NY)                                    INTRPB79
      GO TO 140                                                         INTRPB80
130   WRITE (6,170) Z,(ZT(I),I=1,NZ)                                    INTRPB81
140   STOP                                                              INTRPB82
C                                                                       INTRPB83
C                                                                       INTRPB84
150   FORMAT (38H FIRST ARGUMENT IS OUTSIDE TABLE,  X =,E13.6/5H XT =,/(INTRPB85
     11X,5E13.6))                                                       INTRPB86
160   FORMAT (39H SECOND ARGUMENT IS OUTSIDE TABLE,  Y =,E13.6/5H YT =,/INTRPB87
     1(1X,5E13.6))                                                      INTRPB88
170   FORMAT (38H THIRD ARGUMENT IS OUTSIDE TABLE,  Z =,E13.6/5H ZT =,/(INTRPB89
     11X,5E13.6))                                                       INTRPB90
      END                                                               INTRPB91
```

AD-A106 520 DOUGLAS AIRCRAFT CO LONG BEACH CA F/G 18/3
NUCLEAR BLAST RESPONSE COMPUTER PROGRAM. VOLUME III. PROGRAM LI--ETC(U)
AUG 81 J A MCGREW, H H CROXEN, T P KALMAN DNA001-75-C-0216
UNCLASSIFIED AFWL-TR-81-32-VOL-3 NL

5 of 5
AD
A106520

END
DATE
FILMED
12-81
DTIC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

```
      SUBROUTINE LOADCH(PLOT,ITERS,                                       LOADCH 2
     1 NINTLD,NTMRSP,NGL,NGLL,GAMX,GAMY,GAMZ,INDSYM,                      LOADCH 3
     1 PKL,PST,PAT,PSTR,PATR,STALDS,TIME,                                 LOADCH 4
     1 VGUST,VGUSTR,PO,DELPR,DELPRR,VSS)                                  LOADCH 5
C                                                                         LOADCH 6
      COMMON/XTF/NF(100)                                                  LOADCH 7
      EQUIVALENCE  (NF(25),KPRBLS)                                        LOADCH 8
     1 , (NF(28),IPLBL)                                                   LOADCH 9
C                                                                         LOADCH10
      COMMON/ZZZ/CASF(48),NIN,NOUT,KROW,LINES,IPRNT,NER                   LOADCH11
C                                                                         LOADCH12
      DIMENSION PKL(NINTLD,8)                                             LOADCH13
      DIMENSION PST(NINTLD,1),PAT(NINTLD,1),PSTR(1),PATR(1)               LOADCH14
      DIMENSION STALDS(NINTLD,8),TIME(1)                                  LOADCH15
      DIMENSION PLOT(1)                                                   LOADCH16
C                                                                         LOADCH17
C                                                                         LOADCH18
    1 FORMAT(1H0,20X, LOAD TIME HISTORIES /1H0,                           LOADCH19
     1 20X, ORIENTATION NO. ,I3,  GAMX =  ,E12.5,  GAMY =  ,E12.5,        LOADCH20
     1   GAMZ =  ,E12.5)                                                  LOADCH21
    2 FORMAT(1H ,I4,2X,F8.4,8(1X,E12.5))                                  LOADCH22
    3 FORMAT(1H0,  NO ,6X, TIME ,4(5X, R-STA ,I3,5X, L-STA ,I3))          LOADCH23
    4 FORMAT(1H0,20X, MAXIMUM LOADS /1H0,  STA ,6X, TIME ,                LOADCH24
     1 6X, R SIDE+ ,5X, TIME ,6X, R SIDE- ,5X, TIME ,6X, L SIDE+ ,        LOADCH25
     1 5X, TIME ,6X, L SIDE- ,8X, MAX+ ,8X, MAX- )                        LOADCH26
    5 FORMAT(1H ,I4,1X,4(1X,F8.4,1X,E12.5), 2(1X,E12.5))                  LOADCH27
    6 FORMAT(1H0,20X, ORIENTATION NO. ,I4)                                LOADCH28
   11 FORMAT(1H0,20X, LOAD TIME HISTORY FOR RHS LOAD NO. ,I4,             LOADCH29
     1     ORIENTATION NO. ,I4)                                           LOADCH30
   12 FORMAT(1H0,20X, LOAD TIME HISTORY FOR LHS LOAD NO. ,I4,             LOADCH31
     1     ORIENTATION NO. ,I4)                                           LOADCH32
    7 FORMAT(1H0,20X, MATERIAL VELOCITY= ,F10.4, FPS /1H ,                LOADCH33
     1 20X, MAX ALLOWABLE VELOCITY= ,F10.4, FPS /1H ,20X, P AMBIENT= ,    LOADCH34
     1 F10.4,  PSI /1H ,20X, OVERPRESSURE= ,F10.4,  PSI /1H ,20X,         LOADCH35
     1   MAX ALLOW. OVERPRESSURE= ,F10.4,  PSI )                          LOADCH36
C                                                                         LOADCH37
      ITERS=0                                                             LOADCH38
C                                                                         LOADCH39
C     HERE FIND PROPER DIRECTIONS FOR INCREMENTAL LOADS (TRIM DEPENDENT)LOADCH40
      AMS=1.0                                                             LOADCH41
      AMA=1.0                                                             LOADCH42
      IF(NGLL.LT.0)AMA=-1.0                                               LOADCH43
C                                                                         LOADCH44
C     TRIM LOADS ARE IN PSTR,PATR                                         LOADCH45
C     INCREMENTAL LOADS ARE IN PST,PAT                                    LOADCH46
C     TOTAL LOADS ARE IN PL,PA (LHS , RHS)                                LOADCH47
C                                                                         LOADCH48
      DO 500 I=1,NINTLD                                                   LOADCH49
      DO 490 J=1,8                                                        LOADCH50
  490 PKL(I,J)=0.                                                         LOADCH51
C                                                                         LOADCH52
      IF(INDSYM.EQ.1)GO TO 300                                            LOADCH53
```

```
C     CALC LHS RHS LOADS (UNSYMMETRIC LOAD CONDITION)
C
      DO 200 J=1,NTMRSP
      TEMP=PSTR(I)-PATR(I)+AMS*PST(I,J)-AMA*PAT(I,J)
      PST(I,J)=PSTR(I)+PATR(I)+AMS*PST(I,J)+AMA*PAT(I,J)
      PAT(I,J)=TEMP
C     FIND PEAK LHS AND RHS + AND - LOADS
      IF(PST(I,J).LT.PKL(I,2))GO TO 201
      PKL(I,2)=PST(I,J)
      PKL(I,1)=TIME(J)
      GO TO 202
  201 IF(PST(I,J).GT.PKL(I,4))GO TO 202
      PKL(I,4)=PST(I,J)
      PKL(I,3)=TIME(J)
  202 IF(PAT(I,J).LT.PKL(I,6))GO TO 203
      PKL(I,6)=PAT(I,J)
      PKL(I,5)=TIME(J)
      GO TO 204
  203 IF(PAT(I,J).GT.PK (I,8))GO TO 204
      PKL(I,8)=PAT(I,J)
      PKL(I,7)=TIME(J)
  204 CONTINUE
  200 CONTINUE
C
      GO TO 495
C
  300 CONTINUE
C     CALC LHS RHS LOADS (SY-METRIC CONDITION)
C
      DO 350 J=1,NTMRSP
      TEMP=PSTR(I)+AMS*PST(I,J)-PATR(I)
      PST(I,J)=PSTR(I)+AMS*PST(I,J)+PATR(I)
      PAT(I,J)=TEMP
C     FIND PEAK LHS AND RHS + AND - LOADS
      IF(PST(I,J).LT.PKL(I,2))GO TO 301
      PKL(I,2)=PST(I,J)
      PKL(I,1)=TIME(J)
      GO TO 302
  301 IF(PST(I,J).GT.PKL(I,4))GO TO 302
      PKL(I,4)=PST(I,J)
      PKL(I,3)=TIME(J)
  302 IF(PAT(I,J).LT.PKL(I,6))GO TO 303
      PKL(I,6)=PAT(I,J)
      PKL(I,5)=TIME(J)
      GO TO 304
  303 IF(PAT(I,J).GT.PKL(I,8))GO TO 304
      PKL(I,8)=PAT(I,J)
      PKL(I,7)=TIME(J)
  304 CONTINUE
  350 CONTINUE
C
C
```

```
  495 CONTINUE                                                    LOADC106
  500 CONTINUE                                                    LOADC107
C                                                                 LOADC108
C                                                                 LOADC109
      IF(KPRBLS.EQ.0)GO TO 600                                    LOADC110
      CALL HEADNG                                                 LOADC111
      WRITE(NOUT,1)NGL,GAMX,GAMY,GAMZ                             LOADC112
      JF=0                                                        LOADC113
      LINES=4                                                     LOADC114
C                                                                 LOADC115
  550 JS=JF+1                                                     LOADC116
      JF=JS+3                                                     LOADC117
      IF(JF.GT.NINTLD) JF=NINTLD                                  LOADC118
      WRITE(NOUT,3)(J,J,J=JS,JF)                                  LOADC119
      DO 520 I=1,NTMRSP                                           LOADC120
      LINES=LINES+1                                               LOADC121
      IF(LINES.LE.KROW)GO TO 515                                  LOADC122
      CALL HEADNG                                                 LOADC123
      WRITE(NOUT,1)NGL,GAMX,GAMY,GAMZ                             LOADC124
      WRITE(NOUT,3)(J,J,J=JS,JF)                                  LOADC125
      LINES=LINES+4                                               LOADC126
C                                                                 LOADC127
  515 CONTINUE                                                    LOADC128
  520 WRITE(NOUT,2)I,TIME(I),(PST(J,I),PAT(J,I),J=JS,JF)          LOADC129
      IF(JF.GE.NINTLD)GO TO 600                                   LOADC130
      GO TO 550                                                   LOADC131
  600 CONTINUE                                                    LOADC132
C                                                                 LOADC133
      IF(IPLBL.EQ.0)GO TO 630                                     LOADC134
C                                                                 LOADC135
C     NOW PLOT LOAD TIME HIST (RHS)                               LOADC136
C                                                                 LOADC137
      DO 620 I=1,NINTLD                                           LOADC138
      IF(IPLBL.EQ.2)GO TO 625                                     LOADC139
      DO 610 J=1,NTMRSP                                           LOADC140
  610 PLOT(J)=PST(I,J)                                            LOADC141
      CALL PLT1(NTMRSP,TIME,PLOT,1)                               LOADC142
      WRITE(NOUT,11)I,NGL                                         LOADC143
C                                                                 LOADC144
  625 CONTINUE                                                    LOADC145
      IF(IPLBL.EQ.1)GO TO 620                                     LOADC146
      IF(IPLBL.EQ.3.AND.INDSYM.EQ.1)GO TO 620                     LOADC147
C                                                                 LOADC148
      DO 626 J=1,NTMRSP                                           LOADC149
  626 PLOT(J)=PAT(I,J)                                            LOADC150
      CALL PLT1(NTMRSP,TIME,PLOT,1)                               LOADC151
      WRITE(NOUT,12)I,NGL                                         LOADC152
C                                                                 LOADC153
  620 CONTINUE                                                    LOADC154
C                                                                 LOADC155
  630 CONTINUE                                                    LOADC156
C                                                                 LOADC157
```

```
C                                                                       LOADC158
      CALL HEADNG                                                       LOADC159
      WRITE(NOUT,4)                                                     LOADC160
      PKP=0.                                                            LOADC161
      PKM=0.                                                            LOADC162
      IIP=0                                                             LOADC163
      IIM=0                                                             LOADC164
      DO 650 I=1,NINTLD                                                 LOADC165
      LINES=LINES+1                                                     LOADC166
      IF(LINES.LT.KROW)GO TO 642                                        LOADC167
      CALL HEADNG                                                       LOADC168
      WRITE(NOUT,4)                                                     LOADC169
  642 CONTINUE                                                          LOADC170
      WRITE(NOUT,5)I,(PKL(I,J),J=1,8),(STALDS(I,J),J=7,8)               LOADC171
C     FIND MAX POS LOAD RATIO                                           LOADC172
      DO 640 J=2,6,4                                                    LOADC173
      IF(STALDS(I,7).EQ.0)GO TO 640                                     LOADC174
      RATIO=PKL(I,J)/STALDS(I,7)                                        LOADC175
      IF(RATIO.LT.PKP)GO TO 640                                         LOADC176
      PKP=RATIO                                                         LOADC177
      IIP=I                                                             LOADC178
      TIMP=PKL(I,J-1)                                                   LOADC179
  640 CONTINUE                                                          LOADC180
C     FIND MAX NEG LOAD RATIO                                           LOADC181
      DO 645 J=4,8,4                                                    LOADC182
      IF(STALDS(I,8).EQ.0)GO TO 645                                     LOADC183
      RATIO=PKL(I,J)/STALDS(I,8)                                        LOADC184
      IF(RATIO.LT.PKM)GO TO 645                                         LOADC185
      PKM=RATIO                                                         LOADC186
      IIM=I                                                             LOADC187
      TIMM=PKL(I,J-1)                                                   LOADC188
  645 CONTINUE                                                          LOADC189
C                                                                       LOADC190
  650 CONTINUE                                                          LOADC191
C                                                                       LOADC192
C     FIND NEW ALLOW GUST VEL                                           LOADC193
      PKPR=1.0                                                          LOADC194
      PKMR=1.0                                                          LOADC195
      IF(IIP.EQ.0)GO TO 660                                             LOADC196
      PKPR=PKP                                                          LOADC197
  660 IF(IIM.EQ.0)GO TO 670                                             LOADC198
      PKMR=PKM                                                          LOADC199
  670 CONTINUE                                                          LOADC200
      PKMAX = AMAX1(PKPR,PKMR)                                          LOADC201
      IF(ABS(1.-PKMAX).LE.0.025)ITERS=1                                 LOADC202
      VGUSTR=VGUST/PKMAX                                                LOADC203
      DELP=PO*DELPR                                                     LOADC204
      RAT=VGUSTR/VSS                                                    LOADC205
      FCT=(21./25.)*RAT**2                                              LOADC206
      DELPRR=FCT+SQRT(FCT**2+(49./25.)*RAT**2)                          LOADC207
      DELPP=PO*DELPRR                                                   LOADC208
      WRITE(NOUT,6)NGL                                                  LOADC209
```

```
      WRITE(NOUT,7)VGUST,VGUSTR,PO,DELP,DELPP                           LOADC210
      WRITE(NOUT,10)IIP,PKPR,TIMP,IIM,PKMR,TIMM                         LOADC211
   10 FORMAT(1H0,20X, MAXIMUM POS. AND NEG. LOAD RATIUS /1H0,           LOADC212
     1 20X, STA ,4X, LOAD ,5X, TIME /1H ,                               LOADC213
     1 20X,I3,2(1X,F8.4)/1H ,20X,I3,2(1X,F8.4))                         LOADC214
C                                                                       LOADC215
C                                                                       LOADC216
      RETURN                                                            LOADC217
      END                                                               LOADC218
```

```
      SUBROUTINE PRESS(R,Z)                                             PRESS   2
C     GIVEN SCALED RANGE, R, SUBROUTINE PRESS RETURNS THE SHOCK         PRESS   3
C     OVERPRESSURE RATIO IN Z.                                          PRESS   4
      DIMENSION RS(7), A(7), B(7), C(7)                                 PRESS   5
      DATA RS/2.4,1.4,1.0,0.8,0.6,0.4,0.3/,A/0.36685,0.84705,1.79828,3.3PRESS   6
     10462,5.70575,14.48829,33.02947/,B/-0.134477,-0.549155,-1.870318,-4PRESS   7
     2.82975,-11.04581,-41.91963,-134.76857/,C/0.014838,0.104254,0.56261PRESS   8
     38,2.01571,6.03402,33.09444,149.33439/                             PRESS   9
      IF (R.LT.3.0) GO TO 20                                            PRESS  10
      Z=0.37675/R**1.2354                                               PRESS  11
   10 RETURN                                                            PRESS  12
   20 DO 30 I=1,7                                                       PRESS  13
      IF (R.GE.RS(I)) GO TO 50                                          PRESS  14
   30 CONTINUE                                                          PRESS  15
      IF (R.LT.0.115) GO TO 40                                          PRESS  16
      Z=0.27889/R**2.5542                                               PRESS  17
      GO TO 10                                                          PRESS  18
   40 WRITE (6,60) R                                                    PRESS  19
      STOP                                                              PRESS  20
   50 Z=(C(I)*R+B(I))*R+A(I)                                            PRESS  21
      GO TO 10                                                          PRESS  22
C                                                                       PRESS  23
C                                                                       PRESS  24
   60 FORMAT (17H SCALED RANGE OF ,E11.4,55H KILOFEET IS LESS THAN MINIMPRESS  25
     1UM PERMISSIBLE VALUE, 0.115)                                      PRESS  26
      END                                                               PRESS  27
```

```
      SUBROUTINE RANGE(P,R)
C
      COMMON/ZZZ/CASE(48),NIN,NOUT,KROW,LINES,IPRNT,NER
C
C     GIVEN OVERPRESSURE RATIO P, SCALED RANGE R IS RETURNED
C
      DIMENSION PT(7),A(7),B(7),C(7)
C
      DATA PT/0.12957,0.28257,0.49058,0.73087,1.25051,
     1 3.01555,6.03899/
C
      DATA A/4.531507,2.636265,1.662156,1.198027,0.915294,
     1 0.633333,0.451231/
C
      DATA B/67.394528,9.601168,1.777405,0.496103,0.165727,
     1 0.030217,0.006696/
C
      DATA C/-24.723682,-8.132669,-3.196272,-1.639432,
     1 -0.945597,-0.437786,-0.221178/
C
C
      IF(P.GT.0.09697)GO TO 20
      R=(0.37675/P)**0.809454
      GO TO 100
C
   20 DO 30 I=1,7
      IF(P.LE.PT(I))GO TO 50
   30 CONTINUE
C
      IF(P.GT.69.9195)GO TO 40
      R=(0.27889/P)**0.391512
      GO TO 100
C
   40 WRITE(NOUT,60)P
   60 FORMAT(1H1,20X, OVERPRESSURE RATIO OF   ,E12.5,
     1    IS GREATER THAN MAX ALLOWABLE )
      STOP
C
   50 R=A(I)-SQRT(A(I)**2+B(I)*P+C(I))
C
  100 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE STRSCH(                                                STRSCH 2
     1 ITERS,NINTLD,NTMRSP,                                             STRSCH 3
     1 PKL,PST,PAT,STL,STR,ALLOWS,TIME,                                 STRSCH 4
     1 VGUST,VGUSTR,PO,DELPR,DELPRR,VSS,                                STRSCH 5
     1 STRESS,NSTRSS,NCRITS)                                            STRSCH 6
C                                                                       STRSCH 7
      COMMON/777/CASE(48),NIN,NOUT,KROW,LINES,IPRNT,NER                 STRSCH 8
C                                                                       STRSCH 9
      COMMON/XTF/NF(100)                                                STRSCH10
      EQUIVALENCE (NF(25),KPRBLS)                                       STRSCH11
C                                                                       STRSCH12
      DIMENSION PKL(NSTRSS,8)                                           STRSCH13
      DIMENSION PST(NINTLD,1),PAT(NINTLD,1)                             STRSCH14
      DIMENSION ALLOWS(NSTRSS,2),TIME(1)                                STRSCH15
      DIMENSION STL(NSTRSS,1),STR(NSTRSS,1)                             STRSCH16
      DIMENSION STRESS(NSTRSS,1)                                        STRSCH17
C                                                                       STRSCH18
C                                                                       STRSCH19
    1 FORMAT(1HO,20X, STRESS TIME HISTORIES )                           STRSCH20
    2 FORMAT(1H ,I4,2X,F8.4,8(1X,E12.5))                                STRSCH21
    3 FORMAT(1HO,   NO ,6X, TIME ,4(5X, R-STA ,I3,5X, L-STA ,I3))       STRSCH22
    4 FORMAT(1HO,20X, MAXIMUM STRESSES /1HO,  STA ,6X, TIME ,           STRSCH23
     1 6X, R SIDE+ ,5X, TIME ,6X, R SIDE- ,5X, TIME ,6X, L SIDE+ ,      STRSCH24
     1 5X, TIME ,6X, L SIDE- ,8X, MAX+ ,8X, MAX- )                      STRSCH25
    5 FORMAT(1H ,I4,1X,4(1X,F8.4,1X,E12.5),2(1X,E12.5))                 STRSCH26
    7 FORMAT(1HO,20X, MATERIAL VELOCITY= ,F10.4, FPS /1H ,              STRSCH27
     1 20X, MAX ALLOWABLE VELOCITY= ,F10.4, FPS /1H ,20X, P AMBIENT= ,  STRSCH28
     1 F10.4,  PSI /1H ,20X, OVERPRESSURE= ,F10.4,  PSI /1H ,20X,       STRSCH29
     1    MAX ALLOW. OVERPRESSURE= ,F10.4,  PSI )                       STRSCH30
C                                                                       STRSCH31
C                                                                       STRSCH32
      IF(NCRITS.NE.0)ITERS=0                                            STRSCH33
C                                                                       STRSCH34
      DO 500 I=1,NSTRSS                                                 STRSCH35
      DO 490 J=1,8                                                      STRSCH36
  490 PKL(I,J)=0.                                                       STRSCH37
C                                                                       STRSCH38
      DO 200 J=1,NTMRSP                                                 STRSCH39
      STL(I,J)=0.                                                       STRSCH40
      STR(I,J)=0.                                                       STRSCH41
C                                                                       STRSCH42
      DO 150 K=1,NINTLD                                                 STRSCH43
      STL(I,J)=STL(I,J)+STRESS(I,K)*PAT(K,J)                            STRSCH44
  150 STR(I,J)=STR(I,J)+STRESS(I,K)*PST(K,J)                            STRSCH45
C                                                                       STRSCH46
C     FIND PEAK LHS AND RHS + AND - STRESS                              STRSCH47
      IF(STR(I,J).LT.PKL(I,2))GO TO 201                                 STRSCH48
      PKL(I,2)=STR(I,J)                                                 STRSCH49
      PKL(I,1)=TIME(J)                                                  STRSCH50
      GO TO 202                                                         STRSCH51
  201 IF(STR(I,J).GT.PKL(I,4))GO TO 202                                 STRSCH52
      PKL(I,4)=STR(I,J)                                                 STRSCH53
```

```
      PKL(I,3)=TIME(J)                                                  STRSCH54
  202 IF(STL(I,J).LT.PKL(I,6))GO TO 203                                 STRSCH55
      PKL(I,6)=STL(I,J)                                                 STRSCH56
      PKL(I,5)=TIME(J)                                                  STRSCH57
      GO TO 204                                                         STRSCH58
  203 IF(STL(I,J).GT.PKL(I,8))GO TO 204                                 STRSCH59
      PKL(I,8)=STL(I,J)                                                 STRSCH60
      PKL(I,7)=TIME(J)                                                  STRSCH61
  204 CONTINUE                                                          STRSCH62
  200 CONTINUE                                                          STRSCH63
C                                                                       STRSCH64
 500  CONTINUE                                                          STRSCH65
C                                                                       STRSCH66
C                                                                       STRSCH67
      IF(KPPBLS.EQ.0)GO TO 390                                          STRSCH68
      CALL HEADNG                                                       STRSCH69
      WRITE(NOUT,1)                                                     STRSCH70
      JF=0                                                              STRSCH71
      LINES=2                                                           STRSCH72
C                                                                       STRSCH73
  550 JS=JF+1                                                           STRSCH74
      JF=JS+3                                                           STRSCH75
      IF(JF.GT.NSTRSS) JF=NSTRSS                                        STRSCH76
      WRITE(NOUT,3)(J,J,J=JS,JF)                                        STRSCH77
      DO 520 I=1,NTMRSP                                                 STRSCH78
      LINES=LINES+1                                                     STRSCH79
      IF(LINES.LE.KROW)GO TO 300                                        STRSCH80
      CALL HEADNG                                                       STRSCH81
      WRITE(NOUT,1)                                                     STRSCH82
      WRITE(NOUT,3)(J,J,J=JS,JF)                                        STRSCH83
      LINES=LINES+2                                                     STRSCH84
  300 CONTINUE                                                          STRSCH85
  520 WRITE(NOUT,2)I,TIME(I),(STR(J,I),STL(J,I),J=JS,JF)                STRSCH86
      IF(JF.GE.NSTRSS)GO TO 600                                         STRSCH87
      GO TO 550                                                         STRSCH88
  600 CONTINUE                                                          STRSCH89
C                                                                       STRSCH90
  390 CONTINUE                                                          STRSCH91
C                                                                       STRSCH92
      CALL HEADNG                                                       STRSCH93
      WRITE(NOUT,4)                                                     STRSCH94
      IIP=0                                                             STRSCH95
      IIM=0                                                             STRSCH96
      PKP=0.                                                            STRSCH97
      PKM=0.                                                            STRSCH98
      DO 650 I=1,NSTRSS                                                 STRSCH99
      WRITE(NOUT,5)I,(PKL(I,J),J=1,8)                                   STRSC100
      DO 640 J=2,6,4                                                    STRSC101
      IF(ALLOWS(I,1).EQ.0)GO TO 640                                     STRSC102
      RATIO=PKL(I,J)/ALLOWS(I,1)                                        STRSC103
      IF(RATIO.LT.PKP)GO TO 640                                         STRSC104
      PKP=RATIO                                                         STRSC105
```

```
      IIP=I
  640 CONTINUE
      DO 645 J=4,8,4
      IF(ALLOWS(I,2).EQ.0)GO TO 645
      RATIO=PKL(I,J)/ALLOWS(I,2)
      IF(RATIO.LT.PKM)GO TO 645
      PKM=RATIO
      IIM=I
  645 CONTINUE
C
  650 CONTINUE
C
      IF(NCRITS.EQ.0)GO TO 1000
C     FIND NEW ALLOW GUST VEL
C
      PKPR=1.0
      PKMR=1.0
      IF(IIP.EQ.0)GO TO 660
      PKPR=PKP
  660 IF(IIM.EQ.0)GO TO 670
      PKMR=PKM
  670 CONTINUE
      PKMAX=AMAX1(PKPR,PKMR)
      IF(ABS(1.-PKMAX).LE.0.025)ITERS=1
      VGUSTR=VGUST*PKMAX
      DELP=PO*DELPR
      RAT=VGUSTR/VSS
      FCT=(21./25.)*RAT**2
      DELPRR=FCT+SQRT(FCT**2+(49./25.)*RAT**2)
      DELPP=PO*DELPRR
      WRITE(NOUT,7)VGUST,VGUSTR,PO,DELP,DELPP
C
 1000 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE TAR(R,T)
C     GIVEN SCALED RANGE, R, WITH T=0, SUBROUTINE TAR
C     RETURNS THE SCALED TIME OF ARRIVAL OF THE SHOCK WAVE IN T.
C     GIVEN SCALED TIME OF ARRIVAL IN T, WITH R=0, SUBROUTINE TAR
C     RETURNS THE SCALED SHOCK RADIUS IN R.
      DIMENSION RS(5), A(5), B(5), C(5)
      DATA RS/5.867375,1.68,0.43,0.265,0.115/,A/-.632144,-.3803754,-.141
     1976,-0.0162814,-0.003187,B/0.895889,0.810069,0.5227985,0.0349106,-
     20.0450537/,C/0.0,0.00731329,0.0938408,0.548666,0.663855/
      IF (R.EQ.0.0) GO TO 40
      DO 10 I=1,5
      IF (R.GE.RS(I)) GO TO 20
   10 CONTINUE
      WRITE (6,70) R
      STOP
   20 T=(C(I)*R+B(I))*R+A(I)
   30 RETURN
   40 R=(T-A(1))/B(1)
      IF (R.GE.RS(1)) GO TO 30
      DO 50 I=2,5
      TT=(C(I)*RS(I)+B(I))*RS(I)+A(I)
      IF (T.LT.TT) GO TO 50
      R=(-B(I)+SQRT(B(I)**2+4.0*C(I)*(T-A(I))))/(2.0*C(I))
      GO TO 30
   50 CONTINUE
      WRITE (6,60) T
      STOP
C
C
   60 FORMAT (15H SCALED TIME OF ,E11.4,49H IS LESS THAN MINIMUM PERMISS
     1IBLE VALUE, 0.000418)
   70 FORMAT (17H SCALED RANGE OF ,E11.4,55H KILOFEET IS LESS THAN MINIM
     1UM PERMISSIBLE VALUE, 0.115)
      END
```

```
      SUBROUTINE TIMHST(FTG,
     1 NINTLD,NFREQ,NTMRSP,NTMGST,TIMEMX,OMEGMX,INDSYM,RHOA,
     1 VELG,TMGST,RHO,OMEGA,PLOTF,PLOTA,
     2 AS,AA,AST,AAT,NACC,NG,NGL,PLOT,
     1 PS,PA,PST,PAT,TIME)
C
      COMMON/XTF/NF(100)
      EQUIVALENCE (NF(22),KPRBLS)
     1 , (NF(28),IPLBL)
C
      COMMON/ZZZ/CASE(48),NIN,NOUT,KROW,LINES,IPRNT,NER
C
      DIMENSION VELG(1),TMGST(1),RHO(1)
      DIMENSION OMEGA(1),PLOTF(1),PLOTA(1)
      DIMENSION AS(1), AA(1), AST(1), AAT(1), PLOT(1)
      DIMENSION PS(2,NINTLD,1),PA(2,NINTLD,1)
      DIMENSION PST(NINTLD,1),PAT(NINTLD,1)
      DIMENSION TIME(1)
      DIMENSION FTG(2,1)
C
      DATA TWOPI/6.28319/
   10 FORMAT(1H0,20X, SYMMETRIC TIME RESPONSE - INTGD LOADS /1H0,
     1 20X,F8.4,  RESPONSE SECS ,2X,F8.4,  HZ COUNTED )
   11 FORMAT(1H0,20X, ANTISYMMETRIC TIME RESPONSE - INTGD LOADS /1H0,
     1 20X,F8.4,  RESPONSE SECS ,2X,F8.4,  HZ COUNTED )
   20 FORMAT(1H0,20X, SYMMETRIC FREQUENCY RESPONSE FUNCTION )
   15 FORMAT(1H0,20X, ANTISYMMETRIC FREQUENCY RESPONSE FUNCTION )
   25 FORMAT(1H0,2X FREQ   ,4(2X, REAL-STA ,I3,2X, IMAG-STA ,I3))
   40 FORMAT(1H0,2X, TIME   ,8(7X, STA ,I3))
   50 FORMAT(1H ,F8.4,8(1X,E12.5))
C
C
C     INDSYM=1 IS A SYMMETRIC CASE
C     INDSYM=2 IS AN UNSYMMETRIC CASE
C
      IF(IPLBL.EQ.0)GO TO 100
      CALL PLT1(NTMGST,TMGST,VELG,1)
      WRITE(NOUT,5)
    5 FORMAT(1H0,20X, GUST TIME HISTORY AT AAS ORIGIN )
  100 CONTINUE
C
      CALL TREFT(FTG,NTMGST,OMEGA,VELG,RHO,RHOA,NFREQ,PLOTF,PLOTA,
     1 TMGST,GZERO,ALPHA,BETA)
C
      NGLL = IABS(NGL)
C
      IF (NACC.EQ.0) GOTO 150
C
      CALL RDLOAD (6,NGLL,AS,NOUT,NER)
      IF (INDSYM.GT.1) CALL RDLOAD (7,NGLL,AA,NOUT,NER)
      CALL ACCHST
     1 ( FTG,NACC,NFREQ,NTMRSP,OMEGMX,INDSYM,OMEGA
```

```
     2 , AS,AA,AST,AAT,TIME,TIMEMX,KPRBLS,NG,NGL, IPLBL,PLOT)        TIMHST54
C                                                                    TIMHST55
  150 CONTINUE                                                       TIMHST56
      CALL RDLOAD (8,NGLL,PS,NOUT,NER)                               TIMHST57
      IF (INDSYM.GT.1) CALL RDLOAD (9,NGLL,PA,NOUT,NER)              TIMHST58
C                                                                    TIMHST59
      DO 200 J=1,NFREQ                                               TIMHST60
      AR=FTG(1,J)                                                    TIMHST61
      AI=FTG(2,J)                                                    TIMHST62
      DO 200 I=1,NINTLD                                              TIMHST63
      T=PS(1,I,J)*AR-PS(2,I,J)*AI                                    TIMHST64
      PS(2,I,J)=PS(1,I,J)*AI+PS(2,I,J)*AR                            TIMHST65
      PS(1,I,J)=T                                                    TIMHST66
      IF(INDSYM.EQ.1)GO TO 200                                       TIMHST67
      T=PA(1,I,J)*AR-PA(2,I,J)*AI                                    TIMHST68
      PA(2,I,J)=PA(1,I,J)*AI+PA(2,I,J)*AR                            TIMHST69
      PA(1,I,J)=T                                                    TIMHST70
  200 CONTINUE                                                       TIMHST71
C                                                                    TIMHST72
C                                                                    TIMHST73
      IF(KPRBLS.EQ.0)GO TO 295                                       TIMHST74
      LINES=KROW+1                                                   TIMHST75
      JF=0                                                           TIMHST76
  210 JS=JF+1                                                        TIMHST77
      JF=JS+3                                                        TIMHST78
      IF(JF.GT.NINTLD)JF=NINTLD                                      TIMHST79
      IF(LINES.LE.KROW)GO TO 220                                     TIMHST80
      CALL HEADNG                                                    TIMHST81
      WRITE(NOUT,20)                                                 TIMHST82
      WRITE(NOUT,25)(J,J,J=JS,JF)                                    TIMHST83
      LINES=6                                                        TIMHST84
  220 CONTINUE                                                       TIMHST85
      DO 240 K=1,NFREQ                                               TIMHST86
      LINES=LINES+1                                                  TIMHST87
      IF(LINES.LE.KROW)GO TO 230                                     TIMHST88
      CALL HEADNG                                                    TIMHST89
      WRITE(NOUT,20)                                                 TIMHST90
      WRITE(NOUT,25)(J,J,J=JS,JF)                                    TIMHST91
      LINES=6                                                        TIMHST92
  230 CONTINUE                                                       TIMHST93
      FREQ=OMEGA(K)/TWOPI                                            TIMHST94
  240 WRITE(NOUT,50)FREQ,(PS(1,J,K),PS(2,J,K),J=JS,JF)               TIMHST95
      IF(JF.GE.NINTLD)GO TO 250                                      TIMHST96
      J1=JF+1                                                        TIMHST97
      J2=J1+3                                                        TIMHST98
      WRITE(NOUT,25)(J,J,J=J1,J2)                                    TIMHST99
      LINES=LINES+2                                                  TIMHS100
      GO TO 210                                                      TIMHS101
C                                                                    TIMHS102
  250 CONTINUE                                                       TIMHS103
C                                                                    TIMHS104
      IF(INDSYM.EQ.1)GO TO 295                                       TIMHS105
```

```
      LINES=KROW+1                                                      TIMHS106
      JF=0                                                              TIMHS107
  260 JS=JF+1                                                           TIMHS108
      JF=JS+3                                                           TIMHS109
      IF(JF.GT.NINTLD)JF=NINTLD                                         TIMHS110
      IF(LINES.LE.KROW)GO TO 270                                        TIMHS111
      CALL HEADNG                                                       TIMHS112
      WRITE(NOUT,15)                                                    TIMHS113
      WRITE(NOUT,25)(J,J,J=JS,JF)                                       TIMHS114
      LINES=6                                                           TIMHS115
  270 CONTINUE                                                          TIMHS116
      DO 290 K=1,NFREQ                                                  TIMHS117
      LINES=LINES+1                                                     TIMHS118
      IF(LINES.LE.KROW)GO TO 280                                        TIMHS119
      CALL HEADNG                                                       TIMHS120
      WRITE(NOUT,15)                                                    TIMHS121
      WRITE(NOUT,25)(J,J,J=JS,JF)                                       TIMHS122
      LINES=6                                                           TIMHS123
  280 CONTINUE                                                          TIMHS124
      FREQ=OMEGA(K)/TWOPI                                               TIMHS125
  290 WRITE(NOUT,50)FREQ,(PA(1,J,K),PA(2,J,K),J=JS,JF)                  TIMHS126
      IF(JF.GE.NINTLD)GO TO 295                                         TIMHS127
      J1=JF+1                                                           TIMHS128
      J2=J1+3                                                           TIMHS129
      WRITE(NOUT,25)(J,J,J=J1,J2)                                       TIMHS130
      LINES=LINES+2                                                     TIMHS131
      GO TO 260                                                         TIMHS132
C                                                                       TIMHS133
  295 CONTINUE                                                          TIMHS134
C                                                                       TIMHS135
      CALL IFT(NFREQ,OMEGA,OMEGMX,0.,0,NINTLD,PS,                       TIMHS136
     1 NTMRSP,TIME,PST,TIMEMX)                                          TIMHS137
C                                                                       TIMHS138
      IF(INDSYM.NE.1)CALL IFT(NFREQ,OMEGA,OMEGMX,0.,0,NINTLD,PA,        TIMHS139
     1 NTMRSP,TIME,PAT,TIMEMX)                                          TIMHS140
C                                                                       TIMHS141
      FRQMX=OMEGMX/TWOPI                                                TIMHS142
      IF(KPRBLS.EQ.0)GO TO 500                                          TIMHS143
      JF=0                                                              TIMHS144
C                                                                       TIMHS145
      LINES=KROW+1                                                      TIMHS146
C                                                                       TIMHS147
  300 JS=JF+1                                                           TIMHS148
      JF=JS+7                                                           TIMHS149
      IF(JF.GT.NINTLD)JF=NINTLD                                         TIMHS150
      IF(LINES.LE.KROW)GO TO 310                                        TIMHS151
      CALL HEADNG                                                       TIMHS152
      WRITE(NOUT,10)TIMEMX,FRQMX                                        TIMHS153
      WRITE(NOUT,40)(J,J=JS,JF)                                         TIMHS154
      LINES=6                                                           TIMHS155
  310 CONTINUE                                                          TIMHS156
      DO 400 K=1,NTMRSP                                                 TIMHS157
```

```
      LINES=LINES+1                                              TIMHS158
      IF(LINES.LE.KROW)GO TO 315                                 TIMHS159
      CALL HEADNG                                                TIMHS160
      WRITE(NOUT,40)(J,J=JS,JF)                                  TIMHS161
      LINES=6                                                    TIMHS162
  315 CONTINUE                                                   TIMHS163
  400 WRITE(NOUT,50)TIME(K),(PST(J,K),J=JS,JF)                   TIMHS164
      IF(JF.GE.NINTLD)GO TO 410                                  TIMHS165
      J1=JF+1                                                    TIMHS166
      J2=J1+7                                                    TIMHS167
      WRITE(NOUT,40)(J,J=J1,J2)                                  TIMHS168
      LINES=LINES+2                                              TIMHS169
      GO TO 300                                                  TIMHS170
C                                                                TIMHS171
  410 CONTINUE                                                   TIMHS172
      IF(INDSYM.EQ.1)GO TO 500                                   TIMHS173
      JF=0                                                       TIMHS174
C                                                                TIMHS175
      LINES=KROW+1                                               TIMHS176
C                                                                TIMHS177
  420 JS=JF+1                                                    TIMHS178
      JF=JS+7                                                    TIMHS179
      IF(JF.GT.NINTLD)JF=NINTLD                                  TIMHS180
      IF(LINES.LE.KROW)GO TO 320                                 TIMHS181
      CALL HEADNG                                                TIMHS182
      WRITE(NOUT,11)TIMEMX,FRQMX                                 TIMHS183
      WRITE(NOUT,40)(J,J=JS,JF)                                  TIMHS184
      LINES=6                                                    TIMHS185
  320 CONTINUE                                                   TIMHS186
      DO 430 K=1,NTMRSP                                          TIMHS187
      LINES=LINES+1                                              TIMHS188
      IF(LINES.LE.KROW)GO TO 316                                 TIMHS189
      CALL HEADNG                                                TIMHS190
      WRITE(NOUT,40)(J,J=JS,JF)                                  TIMHS191
      LINES=6                                                    TIMHS192
  316 CONTINUE                                                   TIMHS193
  430 WRITE(NOUT,50)TIME(K),(PAT(J,K),J=JS,JF)                   TIMHS194
      IF(JF.GE.NINTLD)GO TO 500                                  TIMHS195
      J1=JF+1                                                    TIMHS196
      J2=J1+7                                                    TIMHS197
      WRITE(NOUT,40)(J,J=J1,J2)                                  TIMHS198
      LINES=LINES+2                                              TIMHS199
      GO TO 420                                                  TIMHS200
C                                                                TIMHS201
  500 CONTINUE                                                   TIMHS202
C                                                                TIMHS203
      RETURN                                                     TIMHS204
      END                                                        TIMHS205
```

```
      SUBROUTINE TPEVAL(
     1 TIME,RHOA,VSS,SLNTRG,SHA,SHB,SGA,SGOTP,KGRD,SRGOTP,CTP,
     1 SCALR,SCALT,GUST,RHO)
C
C     THIS ROUTINE THANKS TO AL SHARP,AFWL AND VIBRA 4
C     SHA=SCALED HEIGHT OBOVE GRD OF REF POINT
C     SHB=SCALED HEIGHT OBOVE GRD OF BURST
C     SGA=SCALED GRD RANGE
C     SGOTP=SCALED GRD RANGE OF START PT OF TPP
C     SRGOTP=SCALED RANGE FROM BURST TO START PT OF TPP
C
      COMMON/ZZZ/HEADR(48),NIN,NOUT
C
      DIMENSION FRRP(6),FRMR(75),SHBT(5),SRT(6),SRMT(15)
C
C     DATA FOR CRFRRR
      DATA FRRP/1.95,1.585,1.515,1.46,1.43,1.4028333/
      DATA SRT/0.0,0.9,1.2,1.6,2.0,2.4/
C     DATA FOR CRFRMR
      DATA SRMT/0.6,0.8,1.0,1.1,1.2,1.4,1.6,2.0,2.2,2.4,3.4,4.4,6.4,50.0
     1 ,500.0/
      DATA SHBT/0.0,0.4,0.7,1.0,1.4/
      DATA FRMR/15*1.2,1.31,1.42,1.49,1.495,1.485,1.460,1.45,1.43
     1 ,1.425,1.42,1.4,1.395,1.39,1.385,1.385,1.33,1.495,1.69,1.745
     2 ,1.765,1.735,1.69,1.64,1.63,1.62,1.58,1.56,1.545,1.54,1.54
     3 ,1.51,1.515,1.535,1.55,1.565,1.605,1.67,1.79,1.82,1.815,1.74
     4 ,1.7,1.66,1.53,1.63,1.525,1.55,1.58,1.6,1.625,1.675,1.735
     5 ,1.845,1.875,1.885,1.79,1.75,1.705,1.675,1.675/
C
C
C
      K=1
      RHO=RHOA
C
      IF(KGRD.EQ.1)GO TO 40
C
   10 K=0
C
   20 RTA=SCALR*SLNTRG
      RR=1.0
      IF(RTA.GT.0.55)RR=1.0+0.06*(RTA-0.55)**(1./3.)
      RP=RTA*RR
      RG=RTA
      SCALTT=SCALT
      FACT=0.
      CALL PRESS(RP,ZETA)
C
   30 CONTINUE
      T=0.
      CALL TAR(RTA,T)
      TMTA=TIME-SCALT*T
      IF(TMTA.LT.-0.10E-04)GO TO 170
```

```
      IF(TMTA.LT.0)TMTA=0.                                              TPEVAL54
C                                                                       TPEVAL55
      CALL GUSRHO (RG,TMTA,ZETA,SCALTT,GUSTR,RHOR)                      TPEVAL56
      GUST=GUSTR*VSS                                                    TPEVAL57
      RHO=RHO+RHOR*RHOA                                                 TPEVAL58
C                                                                       TPEVAL59
      IF(KGRD*K.EQ.0)GO TO 140                                          TPEVAL60
C                                                                       TPEVAL61
      GO TO 60                                                          TPEVAL62
C                                                                       TPEVAL63
   40 CONTINUE                                                          TPEVAL64
      SRIST=0.                                                          TPEVAL65
      T=TIME/SCALT                                                      TPEVAL66
      CALL TAR(SRIST,T)                                                 TPEVAL67
      IF(SRIST.LT.SHB)GO TO 10                                          TPEVAL68
C                                                                       TPEVAL69
      IF(SHA.LT.0)GO TO 150                                             TPEVAL70
C                                                                       TPEVAL71
      IF(SGA.LT.SGOTP)GO TO 50                                          TPEVAL72
C                                                                       TPEVAL73
      SHTP=SGOTP*CTP*(SGA/SGOTP-1.)**1.6                                TPEVAL74
      IF(SHA.LT.SHTP)GO TO 130                                          TPEVAL75
C                                                                       TPEVAL76
   50 CONTINUE                                                          TPEVAL77
      IF(SRIST.LT.SCALR*SLNTRG)GO TO 140                                TPEVAL78
C                                                                       TPEVAL79
      GO TO 20                                                          TPEVAL80
C                                                                       TPEVAL81
   60 CONTINUE                                                          TPEVAL82
      SRRS=SQRT(SGA**2+(SHA+SHB)**2)                                    TPEVAL83
      IF(SRIST.GT.SRGOTP) GO TO 70                                      TPEVAL84
C                                                                       TPEVAL85
      IF(SRIST.LT.SRRS)GO TO 140                                        TPEVAL86
C                                                                       TPEVAL87
      GO TO 80                                                          TPEVAL88
C                                                                       TPEVAL89
   70 CONTINUE                                                          TPEVAL90
      CALL TPINT(SHB,SRIST,CTP,SGOTP,SHINT,SGINT)                       TPEVAL91
      IF(SGINT**2+(SHINT+SHB)**2.LT.SRRS**2)GO TO 140                   TPEVAL92
C                                                                       TPEVAL93
   80 CONTINUE                                                          TPEVAL94
      IF(SRRS.GT.SRGOTP)GO TO 120                                       TPEVAL95
C                                                                       TPEVAL96
      IF(SRRS.GE.2.4)GO TO 90                                           TPEVAL97
C                                                                       TPEVAL98
      CALL INTPPB(SRRS,0.,0.,6,0,0,SRT,0.,0.,FRRR,CRFRRR)               TPEVAL99
      GO TO 100                                                         TPEVAL100
C                                                                       TPEVAL101
   90 CONTINUE                                                          TPEVAL102
      CRFRRR=1.147+0.614/SRRS                                           TPEVAL103
C                                                                       TPEVAL104
  100 CONTINUE                                                          TPEVAL105
```

```
      R=SRRS/CRFRRR                                                    TPEVA106
      CALL PRESS(R,ZETAT)                                              TPEVA107
      RP=SRRS                                                          TPEVA108
      RTA=RP                                                           TPEVA109
      RG=RP                                                            TPEVA110
C                                                                      TPEVA111
  110 CONTINUE                                                         TPEVA112
      SCALTT=SCALT                                                     TPEVA113
      RR=1.0                                                           TPEVA114
      IF(RTA.GT.0.55)RR=1.0+0.06*(RTA-0.55)**(1./3.)                   TPEVA115
      RP=RP*RR                                                         TPEVA116
      CALL PRESS(RP,ZETAI)                                             TPEVA117
      ZETA=ZETAT-ZETAI                                                 TPEVA118
      FACT=2.0                                                         TPEVA119
      K=0                                                              TPEVA120
      GO TO 30                                                         TPEVA121
C                                                                      TPEVA122
  120 CONTINUE                                                         TPEVA123
      SH=-SHB                                                          TPEVA124
      CALL TPINT(SH,SRRS,CTP,SGOTP,SHINT,SGINT)                        TPEVA125
      SRIS=SQRT(SGINT**2+(SHINT-SHB)**2)                               TPEVA126
      SRMS=SQRT(SGINT**2+SHINT**2)                                     TPEVA127
      SH=AMIN1(SHB,1.4)                                                TPEVA128
      CALL INTRPB(SH,SRMS,0.,5,15,0,SHBT,SRMT,0.,FRMR,CRFRMR)          TPEVA129
      R=SRMS/CRFRMR                                                    TPEVA130
      CALL PRESS(R,ZETATM)                                             TPEVA131
      ZETAT=AMIN1(ZETATP,ZETATM)                                       TPEVA132
      RTA=SRIS                                                         TPEVA133
      RP=SRIS                                                          TPEVA134
      RG=SRRS                                                          TPEVA135
      GO TO 110                                                        TPEVA136
C                                                                      TPEVA137
  130 CONTINUE                                                         TPEVA138
      CALL TPINT(SHB,SRIST,CTP,SGOTP,SHINT,SGINT)                      TPEVA139
      SRMS=SQRT(SGA**2+SHA**2)                                         TPEVA140
      IF(SGINT**2+SHINT**2.LT.SRMS**2)GO TO 140                        TPEVA141
C                                                                      TPEVA142
      SH=0.                                                            TPEVA143
      CALL TPINT(SH,SRMS,CTP,SGOTP,SHINT,SGINT)                        TPEVA144
      SRIS=SQRT(SGINT**2+(SHINT-SHB)**2)                               TPEVA145
      SH=AMIN1(SHB,1.4)                                                TPEVA146
      CALL INTRPB(SH,SRMS,0.,5,15,0,SHBT,SRMT,0.,FRMR,CRFRMR)          TPEVA147
      R=SRMS/CRFRMR                                                    TPEVA148
      CALL PRESS(R,ZETA)                                               TPEVA149
      ZETA=AMIN1(ZETATP,ZETA)                                          TPEVA150
      RTA=SRIS                                                         TPEVA151
      RG=SRMS/1.2                                                      TPEVA152
      SCALTT=1.2*SCALT                                                 TPEVA153
      FACT=1.                                                          TPEVA154
      K=0                                                              TPEVA155
      GO TO 30                                                         TPEVA156
C                                                                      TPEVA157
```

```
  140 CONTINUE                                                              TPEVA158
      RETURN                                                                TPEVA159
C                                                                           TPEVA160
  150 CONTINUE                                                              TPEVA161
      WRITE(NOUT,160)                                                       TPEVA162
  160 FORMAT(1H0,20X, AAS ORIGIN BELOW GRD LEVEL - CHECK FLIGHT PLAN )      TPEVA163
      NER=1                                                                 TPEVA164
      RETURN                                                                TPEVA165
C                                                                           TPEVA166
  170 CONTINUE                                                              TPEVA167
      WRITE(NOUT,180)TMTA                                                   TPEVA168
  180 FORMAT(1H0,20X, A GOCF,DELTA T AFTER INTERCEPT IS NEG.= ,E16.8)       TPEVA169
      TIME=TIME+0.0010                                                      TPEVA170
      GO TO 30                                                              TPEVA171
C                                                                           TPEVA172
      END                                                                   TPEVA173
```

```
      SUBROUTINE TPINT (H,R,C,GO,HI,GI)
C     GIVEN H, R, C, GO, SUBROUTINE TPINT RETURNS HI AND GI.
C     H IS THE SCALED HEIGHT OF THE CENTER OF THE CIRCLE
C     WHOSE INTERSECTION WITH THE TRIPLE POINT PATH IS SOUGHT.
C     R IS THE RADIUS OF THE CIRCLE.
C     C IS THE CONSTANT IN THE EQUATION FOR THE HEIGHT OF THE TRIPLE
C     POINT PATH, HT, AS A FUNCTION OF GROUND RANGE,G.
C              HT=C*GO*(G/GO-1.0)**1.6
C     GO IS THE GROUND RANGE AT WHICH THE TRIPLE POINT PATH STARTS.
C     HI IS THE HEIGHT OF THE INTERSECTION.
C     GI IS THE GROUND RANGE OF THE INTERSECTION.
      D=C
10    T=D*(D*GO+H)
      G=(T+SQRT(T**2-(1.0+D**2)*((D*GO+H)**2-R**2)))/(1.0+D**2)
      HT=C*GO*(G/GO-1.0)**1.6
      RT=SQRT(G**2+(HT-H)**2)
      IF (ABS(RT/R-1.0).LE.0.000001) GO TO 20
      D=HT/(G-GO)
      GO TO 10
20    HI=HT
      GI=G
      RETURN
      END
```

```
      SUBROUTINE TRFFT(FTG,
     1 NT,OMEGA,VELG,RHO,RHOA,NFREQ,PLOTF,PLOTA,TIME,GZERO,ALPHA,BETA)
C
      COMMON/ZZZ/CASE(48),NIN,NOUT,KROW,LINES,IPRNT,NER
C
      COMMON/XTF/NF(100)
      EQUIVALENCE (NF(22),IPREQ)
     1 , (NF(1),AMACH),(NF(2),GAMX)
     1 , (NF(91),FINFRQ)
     2 , (NF(89),TAU)
C
C
      DIMENSION OMEGA(1),VELG(1),RHO(1)
      DIMENSION PLOTF(1)
      DIMENSION PLOTA(1)
      DIMENSION TIME(1)
      DIMENSION FTG(2,1)
C
      DATA TWOPI/6.28318/
      DATA EPV/0.05/
C
    1 FORMAT(1HO,20X, TRANSFORMED FREQUENCY SPECTRUM /1HO,
     1 20X,I4,  FREQUENCIES ,I4,  TIME POINTS )
    5 FORMAT(1HO,
     1  NFRQ ,2X,       FREQ ,              REAL ,              IMAG ,
     1                MOD ,2X,
     1  NFRQ ,2X,       FREQ ,              REAL ,              IMAG ,
     1                MOD )
    2 FORMAT(1H ,I4,2X,F8.3,3E14.5,2X,I4,2X,F8.3,3E14.5)
C
      NFREQP=NFREQ/2
C
      II=0
      DO 20 I=1,NT
      VELG(I)=VELG(I)*RHO(I)/RHOA
      IF(II.GT.0)GO TO 20
      IF(VELG(I).LT.0)II=I
   20 CONTINUE
C
      ITEST = 1
      IF  (II.EQ.0) GO TO 30
C
      GZERO=VELG(1)
      T1  =  TIME(II-1) + TAU
      T2  =  TIME(II)  + TAU
      V1=VELG(II-1)
      V2=VELG(II)
      TO  =  TIME(II-1)  + TAU  +  (T2-T1)*V1/(V1-V2)
      V0=0.0
      T1  =  TIME(II/2)  + TAU
      V1=VELG(II/2)
C
```

```
      GAMMA=ALOG(2.0)/TO
      ALPHA=-(ALOG(V1/GZERO)-ALOG(2.0-2.0**(T1/TO)))/T1
      BETA=ALPHA-GAMMA
C
      T1 = T1 - TAU
      TO =  TO - TAU
C     NOW CHECK FIT
      ITEST=0
      DO 25 I=2,NT,2
      VTEST=GZERO*(2.0*EXP(-ALPHA*TIME(I))-EXP(-BETA*TIME(I)))
      IF(ABS((VTEST-VELG(I))/VTEST).GT.EPV)ITEST=1
   25 CONTINUE
C
      IF(ITEST.EQ.0)GO TO 36
C
   30 CONTINUE
      DO 35 J=1,NFREQ
      FTG(1,J)=0.
      FTG(2,J)=0.
      OMEGI=1.0/OMEGA(J)
      ANG1=OMEGA(J)*TIME(1)
      C1=COS(ANG1)
      S1=SIN(ANG1)
      DO 34 I=2,NT
      DEL=1.0/(TIME(I)-TIME(I-1))
      B=(VELG(I)-VELG(I-1))*DEL
      ANG2=OMEGA(J)*TIME(I)
      S2=SIN(ANG2)
      C2=COS(ANG2)
      RR=OMEGI*(VELG(I)*S2-VELG(I-1)*S1)+B*OMEGI*OMEGI*(C2-C1)
      RI=-OMEGI*(VELG(I)*C2-VELG(I-1)*C1)+B*OMEGI*OMEGI*(S2-S1)
      FTG(1,J)=FTG(1,J)+RR
      FTG(2,J)=FTG(2,J)-RI
      C1=C2
      S1=S2
   34 CONTINUE
   35 CONTINUE
      ALPHA=0.0
      BETA=0.0
C
   36 CONTINUE
C
C
      IF(ITEST.NE.0)GO TO 45
      DO 40 J=1,NFREQ
      C = COS(OMEGA(J)*TAU)
      S = SIN(OMEGA(J)*TAU)
      DENA=ALPHA**2+OMEGA(J)**2
      DENB=BETA**2+OMEGA(J)**2
      F1 = (2.*ALPHA/DENA - BETA/DENB)*GZERO
      F2 = (2./DENA-1.0/DENB)*GZERO*OMEGA(J)
      FTG(1,J) = F1*C + F2*S
```

```
      FTG(2,J) = F1*S - F2*C
   40 CONTINUE
C
   45 CONTINUE
C
      A2=SQRT(FTG(1,NFREQ)**2+FTG(2,NFREQ)**2)
      A1=SQRT(FTG(1,NFREQ-5)**2+FTG(2,NFREQ-5)**2)
      FINFRQ=-(A1*OMEGA(NFREQ)-A2*OMEGA(NFREQ-5))/(A2-A1)
      IF (FINFRQ.LT.0.0) FINFRQ = OMEGA(NFREQ) + 100.
C
      IF(IPREQ.EQ.0)GO TO 50
      CALL HEADNG
      WRITE(NOUT,1)NFREQ,NT
      WRITE(NOUT,4)TIME(1),GZERO,T1,V1,T0,V0,TIME(NT),VFLG(NT),
     1 ALPHA,BETA
    4 FORMAT(1H0,24X, TIME ,3X, V*RHO /1H ,20X,F8.4,1X,F8.4/
     1 1H ,20X,F8.4,1X,F8.4/
     1 1H ,20X,F8.4,1X,F8.4/
     1 1H ,20X,F8.4,1X,F8.4/
     1 1H0,20X, ALPHA= ,E13.6,    BETA= ,E13.6)
      WRITE(NOUT,5)
      LINES=13
      DO 60 J=1,NFREQP
      JJ=J+NFREQP
      PLOTF(J )=OMEGA(J )/TWOPI
      PLOTF(JJ)=OMEGA(JJ)/TWOPI
      PLOTA(J)=SQRT(FTG(1,J)**2+FTG(2,J)**2)
      PLOTA(JJ)=SQRT(FTG(1,JJ)**2+FTG(2,JJ)**2)
      IF(LINES.LT.KROW)GO TO 55
      CALL HEADNG
      WRITE(NOUT,1)NFREQ,NT
      WRITE(NOUT,5)
      LINES=7
   55 LINES=LINES+1
      WRITE(NOUT,2)J,PLOTF(J),FTG(1,J),FTG(2,J),PLOTA(J),
     1 JJ,PLOTF(JJ),FTG(1,JJ),FTG(2,JJ),PLOTA(JJ)
   60 CONTINUE
C
   50 CONTINUE
      FINFRF=FINFRQ/TWOPI
C
      WRITE(NOUT,3)FINFRF
    3 FORMAT(1H ,20X, EST. FREQ. FOR 0 MOD. OF GUST FUNCTION = ,
     ' F10.2,  HZ )
C
      RETURN
      END
```

```
      SUBROUTINE BLINT (NBOX,NSBETO,GEOMBX,GEOMBD,GAMX,GAMY,GAMZ
     1, ALI, X, Y, Z )
C
      DIMENSION GEOMBX(1), GEOMBD(1)
C
      DATA EPTHTA/1.0E-02/
C
C     NOW FIND GEOMETRIC LOC OF BLAST INTERCEPTION ON A/C
C
      ALI=1.E+10
C
      DO 30 I1=1,NBOX
      I2=I1+NBOX
      I3=I2+NBOX
      I4=I3+NBOX
      I5=I4+NBOX
      XINT = GEOMBX(I1)
      YINT = GEOMBX(I2)
      ZINT = GEOMBX(I3)
      YINTL=-YINT
      DY = GEOMBX(I4)
      DZ = GEOMBX(I5)
      DDL=SQRT(DY**2+DZ**2)
      SINDIH=DZ/DDL
      COSDIH=DY/DDL
      THETAR=GAMZ*COSDIH-GAMY*SINDIH
      THETAL=GAMZ*COSDIH+GAMY*SINDIH
      IF(ABS(THETAR).LT.EPTHTA)GO TO 25
      ALR=GAMX*XINT+GAMY*YINT+GAMZ*ZINT
      IF(ALR.GT.ALI)GO TO 25
      ALI=ALR
      X=XINT
      Y=YINT
      Z=ZINT
   25 CONTINUE
      IF(ABS(THETAL).LT.EPTHTA)GO TO 30
      ALL=GAMX*XINT+GAMY*YINTL+GAMZ*ZINT
      IF(ALL.GT.ALI)GO TO 30
      ALI=ALL
      X=XINT
      Y=YINTL
      Z=ZINT
   30 CONTINUE
C
      DO 40 I1=1,NSBETO
      I2=I1+NSBETO
      I3=I2+NSBETO
      I4=I3+NSBETO
      I5=I4+NSBETO
      I6=I5+NSBETO
      XINT=(GEOMBD(I1)+GEOMBD(I4))/2.0
      YINT=(GEOMBD(I2)+GEOMBD(I5))/2.0
```

```
      ZINT=(GEOMBD(I3)+GEOMBD(I6))/2.0                                  BLINT  54
      YINTL=-YINT                                                       BLINT  55
      ALR=GAMX*XINT+GAMY*YINT+GAMZ*ZINT                                 BLINT  56
      IF(ALR.GT.ALI)GO TO 35                                            BLINT  57
      ALI=ALR                                                           BLINT  58
      X=XINT                                                            BLINT  59
      Y=YINT                                                            BLINT  60
      Z=ZINT                                                            BLINT  61
   35 CONTINUE                                                          BLINT  62
      ALL=GAMX*XINT+GAMY*YINTL+GAMZ*ZINT                                BLINT  63
      IF(ALL.GT.ALI)GO TO 40                                            BLINT  64
      ALI=ALL                                                           BLINT  65
      X=XINT                                                            BLINT  66
      Y=YINTL                                                           BLINT  67
      Z=ZINT                                                            BLINT  68
   40 CONTINUE                                                          BLINT  69
C                                                                       BLINT  70
      RETURN                                                            BLINT  71
      END                                                               BLINT  72
```

```
      SUBROUTINE CRIGID                                                 CRIGID 2
C                                                                       CRIGID 3
      COMMON NAA,A(1)                                                   CRIGID 4
      COMMON /ZZZ/HEDR(48),NIN,NOUT,KROW,LINES,IPRNT,NER                ZZZ    2
      COMMON /XTF/NF(100)                                               CRIGID 6
      EQUIVALENCE (NF(6),NSYM),      (NF(9),NFREQ),     (NF(22),IPREQ)  CRIGID 7
     1          ,     (NF(41),NK),       (NF(42),NG)                    CRIGID 8
     2          ,     (NF(43),MBOX),     (NF(44),MSBE)                  CRIGID 9
     3          ,     (NF(62),TIMEMX),   (NF(97),DELT)                  CRIGID10
C                                                                       CRIGID11
   10 FORMAT (6I12)                                                     CRIGID12
   20 FORMAT (6F12.0)                                                   CRIGID13
   30 FORMAT (1H1,20X,*RUN DATA FOR RIGID*                              CRIGID14
     1 / 1H0,20X,*NUMBER OF TIME POINTS                 =*,I4           CRIGID15
     2 / 1H ,20X,*NUMBER OF AERO BOXES                  =*,I4           CRIGID16
     3 / 1H ,20X,*NUMBER OF SLENDER BODY ELEMENTS       =*,I4           CRIGID17
     5 / 1H ,20X,*GUST ORIENTATION                      =*,I4           CRIGID18
     6 / 1H ,20X,*PLOT FLAG                             =*,I4           CRIGID19
     7 / 1H ,20X,*CHECK PRINT FLAG                      =*,I4           CRIGID20
     8 / 1H ,20X,*MAX TIME                              =*,E16.6        CRIGID21
     9 / 1H ,20X,*DELTA TIME                            =*,E16.6        CRIGID22
     Z)                                                                 CRIGID23
   40 FORMAT (1H0,20X,*FREQUENCIES FOR RESPONSE SOLUTION*               CRIGID24
     1 / 1H0,*ITM*,6X,*LWR F*,6X,*UPR F*,6X,*DEL F*)                    CRIGID25
   41 FORMAT (1H ,I3,3F10.4)                                            CRIGID26
C                                                                       CRIGID27
      READ (NIN,10) NTMGST, NBOX, NSBE, NCR, IPLOT, IPREQ               CRIGID28
      READ (NIN,20) TIMEMX, DELT                                        CRIGID29
      WRITE (NOUT,30) NTMGST,NBOX,NSBE,NOR,IPLOT,IPREQ,TIMEMX,DELT      CRIGID30
      NFORCE = NBOX + NSBE                                              CRIGID31
C                                                                       CRIGID32
C     READ IN FREQUENCIES FOR RESPONSE SOLUTION                         CRIGID33
C                                                                       CRIGID34
      READ (NIN,10) NFRGR                                               CRIGID35
      I = 0                                                             CRIGID36
      WRITE (NOUT,40)                                                   CRIGID37
      DO 120 N=1,NFRGR                                                  CRIGID38
      READ (NIN,20) F1,F2,DF                                            CRIGID39
      WRITE (NOUT,41) N,F1,F2,DF                                        CRIGID40
      I = I+1                                                           CRIGID41
      A(I) = F1                                                         CRIGID42
  110 I = I+1                                                           CRIGID43
      A(I) = A(I-1) + DF                                                CRIGID44
      IF (A(I).LT.F2) GO TO 110                                         CRIGID45
      IF (A(I).GT.F2) I=I-1                                             CRIGID46
  120 NFREQ = I                                                         CRIGID47
C                                                                       CRIGID48
      NTMRSP=0                                                          CRIGID49
      NT=TIMEMX/DELT+1                                                  CRIGID50
      DELTT=DELT                                                        CRIGID51
      IND=0                                                             CRIGID52
      TI=0.                                                             CRIGID53
```

```
      DO 200 I=1,NT                                                     CRIGID54
      TI=TI+DELTT                                                       CRIGID55
      IF(TI.GT.TIMEMX)GO TO 250                                         CRIGID56
      NTMRSP=NTMRSP+1                                                   CRIGID57
      IF(IND.GT.1)GO TO 200                                             CRIGID58
      IF(TI.LT.0.25)GO TO 200                                           CRIGID59
      IF(TI.LE.1.0)GO TO 150                                            CRIGID60
      IND=2                                                             CRIGID61
      TI=TI-DELTT                                                       CRIGID62
      DELTT=DELTT*2.0                                                   CRIGID63
      TI=TI+DELTT                                                       CRIGID64
      GO TO 200                                                         CRIGID65
  150 IF(IND.EQ.1)GO TO 200                                             CRIGID66
      IND=1                                                             CRIGID67
      TI=TI-DELTT                                                       CRIGID68
      DELTT=DELTT*5.0                                                   CRIGID69
      TI=TI+DELTT                                                       CRIGID70
  200 CONTINUE                                                          CRIGID71
  250 CONTINUE                                                          CRIGID72
      NTMRSP=NTMRSP+1                                                   CRIGID73
      IF(TI-DELTT.LT.TIMEMX.AND.ABS(TI-DELTT-TIMEMX).GT.0.010)          CRIGID74
     1 NTMRSP=NTMRSP+1                                                  CRIGID75
C                                                                       CRIGID76
      L1   = 1                                                          CRIGID77
      L2   = L1   + NFREQ                                               CRIGID78
      L3   = L2   + MAXO(NTMGST,NTMRSP)                                 CRIGID79
      L4   = L3   + NFORCE                                              CRIGID80
      L5   = L4   + 2*NFORCE                                            CRIGID81
      L6   = L5   + 2*NFORCE*NFREQ                                      CRIGID82
      L7   = L6   + NBOX                                                CRIGID83
      L8   = L7   + 2*NSBE                                              CRIGID84
      L20  = L8   + NSYM                                                CRIGID85
      L21  = L20  + 2*NFORCE*NK                                         CRIGID86
      L22  = L21  + NK                                                  CRIGID87
      L23  = L22  + NFORCE*NK                                           CRIGID88
      L9   = L22                                                        CRIGID89
      L10  = L9   + MBOX*5                                              CRIGID90
      L10A= L10  + MSBE*6                                               CRIGID91
      L11  = L9                                                         CRIGID92
      L12  = L9                                                         CRIGID93
      L13  = L6                                                         CRIGID94
      L14  = L13  + NTMGST                                              CRIGID95
      L15  = L14  + NTMGST                                              CRIGID96
      L16  = L15  + NTMGST                                              CRIGID97
      L17  = L16  + NFREQ                                               CRIGID98
      L18  = L17  + MAXO(NFREQ,NTMRSP)                                  CRIGID99
      L19  = L18                                                        CRIGI100
C                                                                       CRIGI101
      CALL RIGID (NTMGST,NBOX,NSBE,NFORCE,NOR,IPLOT,NTMRSP              CRIGI102
     1 ,  A(L1),A(L2),A(L3),A(L4),A(L5)                                 CRIGI103
     2 ,  A(L6),A(L7),A(L8),A(L9),A(L10),A(L10A),A(L11),A(L12),MBOX,MSBE CRIGI104
     3 ,  A(L13),A(L14),A(L15),A(L16)                                   CRIGI105
```

```
     4 ;  A(L17),A(L18),A(L19)                                                  CRIGI106
     5 ;  A(L20),A(L21),A(L22),A(L23) )                                         CRIGI107
C                                                                               CRIGI108
      RETURN                                                                    CRIGI109
      END                                                                       CRIGI110
```

```
      SUBROUTINE GSINT (NFREQ,NFORCE,NK,IPREQ
     1 , OMEGA,CP,FI,VOBWS,RS,TANS)
C
      DIMENSION OMEGA(1)
      COMPLEX CP(NFORCE,1), FI(NFORCE,1)
      DIMENSION  VOBWS(1), RS(1), TANS(1)
C
      COMMON /XTF/NF(100)
      EQUIVALENCE (NF(12),BZERO),    (NF(18),VTFPS),    (NF(100),SIZFCT)
C
      NKI = MINO(NK,10)
      ISETA = 1
      IF (VOBWS(NK).GT.VOBWS(1)) ISETA = -1
      IMO = 0
      OMEGF = VTFPS*12/(BZERO*SIZFCT)
C
      DO 500 K=1,NFREQ
      VOBW = OMEGF/OMEGA(K)
C
C     FIND OUT IF WE NEED NEW COEFS
      IRECG = 0
      TESTV = 1.E6
      DO 10 I=1,NK
      TEST = ABS(VOBWS(I)-VOBW)
      IF (TEST.GT.TESTV) GO TO 10
      TESTV = TEST
      IM = I
   10 CONTINUE
C
      IL = IM-ISETA
      IF (NKI.GT.4) IL=IL-ISETA
      IH = IL+(NKI-1)*ISETA
C
   20 IF (IL.GE.1.AND.ISETA.GT.0) GO TO 30
      IF (IL.LE.NK.AND.ISETA.LT.0) GO TO 30
      IH = IH+ISETA
      IM = IM+ISETA
      IL = IL+ISETA
      GO TO 20
C
   30 IF (IH.LE.NK.AND.ISETA.GT.0) GO TO 40
      IF (IH.GE.1.AND.ISETA.LT.0) GO TO 40
      IH = IH-ISETA
      IM = IM-ISETA
      IL = IL-ISETA
      GO TO 20
C
   40 IF (IM.NE.IMO) IRECG = 1
      IMO = IM
C
      CALL INTRP2 (IZRO,VOBW,CP(1,K)
     1 , VOBWS(IL),FI(1,IL),NFORCE,NKI,RS,TANS,IRECG,IPREQ,K)
```

```
C                                                                GSINT 54
  500 CONTINUE                                                   GSINT 55
      RETURN                                                     GSINT 56
      END                                                        GSINT 57
```

```
      SUBROUTINE INTRP2 (IZRO,VOBW,FGUST
     1 , VOBWI,FI,NM,NK,RS,TANS,IRECG,IPREQ,J)
C
      DIMENSION VOBWI(1)
      DIMENSION FI(2,NM,1)
      DIMENSION RS(NM,1),TANS(NM,1)
      DIMENSION FGUST(2,1)
      DIMENSION COFF(10), RK(10), C(13,10)
C
      COMMON /ZZZ/CASE(48),NIN,NOUT,KROW,LINES,IPRNT,NER
C
      DATA EPRV/0.001/
      DATA EPC /0.1E-5/
C
      DATA EP/1.0E-02/
C
C     NM = TOTAL AERO ELEMENTS = NBOX+2*NSBETO
C
C
   10 FORMAT(1H0,20X, GUST INTERPOLATION ELEMENTS /1H0,
    1  ELEM  NG  NK ,12X, RR ,12X, RI ,11X, MOD ,11X, TAN )
   20 FORMAT(1H ,3I4,4E14.6)
C
      DO 30 I=1,NK
      K = I
      IF (ABS(1.-VOBWI(I)/VOBW).LE.EPRV) GO TO 100
      COFF(I) = 0.0
   30 RK(I) = 1.0/VOBWI(I)
      IF (IRECG.EQ.1) CALL FORMC (RK,NK,C,IPREQ)
      RKIN = 1.0/VOBW
      CALL COFFF (RKIN,RK,NK,C,COFF)
      DO 35 L=1,NK
   35 IF (ABS(COEF(L)).LT.EPC) COEF(L)=0.0
   39 CONTINUE
C
      K=0
C
      DO 40 I=1,NK
      IF(COEF(I).EQ.1.0)K=I
   40 CONTINUE
C
      IF(K.NE.0)GO TO 100
C
C
C
      IF (IRECG.EQ.0) GO TO 120
C
      ICNT=0
C     SET UP COUNT FOR ZERO GUST LOADS
C
      IF(IPREQ.EQ.0)GO TO 47
C
```

```
      CALL HEADNG                                                       INTRP254
      WRITE(NOUT,10)                                                    INTRP255
C                                                                       INTRP256
   47 CONTINUE                                                          INTRP257
C                                                                       INTRP258
      DO 90 I=1,NM                                                      INTRP259
      DO 80 K=1,NK                                                      INTRP260
      RR=FI(1,I,K)                                                      INTRP261
      RI=FI(2,I,K)                                                      INTRP262
      IF(ABS(RR).LT.EP)RR=0.                                            INTRP263
      IF(ABS(RI).LT.EP)RI=0.                                            INTRP264
      RT=SQRT(RR**2+RI**2)                                              INTRP265
      RS(I,K)=RR                                                        INTRP266
      TANS(I,K)=RI                                                      INTRP267
      IF(IPREQ.NE.0)WRITE(NOUT,20)I,J,K,RR,RI,RT                        INTRP268
      IF(RR.NE.0)GO TO 80                                               INTRP269
      IF(RI.NE.0)GO TO 80                                               INTRP270
      IF(K.GE.2)GO TO 85                                                INTRP271
      GO TO 80                                                          INTRP272
C                                                                       INTRP273
   80 CONTINUE                                                          INTRP274
C                                                                       INTRP275
      GO TO 90                                                          INTRP276
   85 CONTINUE                                                          INTRP277
      DO 86 K=1,NK                                                      INTRP278
      TANS(I,K)=0.                                                      INTRP279
   86 RS(I,K)=0.                                                        INTRP280
      ICNT=ICNT+NK                                                      INTRP281
C                                                                       INTRP282
C                                                                       INTRP283
   90 CONTINUE                                                          INTRP284
      IRECG=1                                                           INTRP285
C                                                                       INTRP286
      IF(ICNT.GE.NK*NM)IZRO=1                                           INTRP287
C                                                                       INTRP288
      IF(IZRO.EQ.1)GO TO 150                                            INTRP289
      GO TO 120                                                         INTRP290
C                                                                       INTRP291
  100 CONTINUE                                                          INTRP292
      DO 110 I=1,NM                                                     INTRP293
C                                                                       INTRP294
      FREAL=FI(1,I,K)                                                   INTRP295
      FIMAG=FI(2,I,K)                                                   INTRP296
      FGUST(1,I)=FREAL                                                  INTRP297
      FGUST(2,I)=FIMAG                                                  INTRP298
C                                                                       INTRP299
  110 CONTINUE                                                          INTRP100
C                                                                       INTRP101
      GO TO 150                                                         INTRP102
C                                                                       INTRP103
  120 CONTINUE                                                          INTRP104
      DO 140 I=1,NM                                                     INTRP105
```

```
      R=0.                                                              INTRP106
      THETA=0.                                                          INTRP107
      DO 130 K=1,NK                                                     INTRP108
      R=R+COEF(K)*RS(I,K)                                               INTRP109
  130 THETA=THETA+COEF(K)*TANS(I,K)                                     INTRP110
C                                                                       INTRP111
      FREAL=R                                                           INTRP112
      FIMAG=THETA                                                       INTRP113
      FGUST(1,I)=FREAL                                                  INTRP114
      FGUST(2,I)=FIMAG                                                  INTRP115
C                                                                       INTRP116
  140 CONTINUE                                                          INTRP117
C                                                                       INTRP118
  150 CONTINUE                                                          INTRP119
C                                                                       INTRP120
      IF(IPREQ.EQ.0)GO TO 160                                           INTRP121
      WRITE (NOUT,36) RKIN                                              INTRP122
      WRITE (NOUT,37) (   RK(K),K=1,NK)                                 INTRP123
      WRITE (NOUT,38) (COEF(K),K=1,NK)                                  INTRP124
   36 FORMAT (1H0,*INTERPOLATION COEFFICIENTS FOR K=*,F16.5)            INTRP125
   37 FORMAT (1H ,*    K=*,10F10.4)                                     INTRP126
   38 FORMAT (1H ,*COEF=*,10F10.4)                                      INTRP127
C                                                                       INTRP128
  160 CONTINUE                                                          INTRP129
C                                                                       INTRP130
      RETURN                                                            INTRP131
C                                                                       INTRP132
      END                                                               INTRP133
```

```
      SUBROUTINE RIGID (NTMGST,NBOX,NSBE,NFORCE,NOP,IPLOT,NTMRSP         RIGID   2
     1 , OMEGA,TIME,GEOM,CPT,CP                                          RIGID   3
     2 , IBOXES,ISBES,Q,GEOP,GEOS,CR,D,DB,MBOX,MSBE                      RIGID   4
     3 , DP,RHO,VG,PLOTF,PLOTA,FTG,P                                     RIGID   5
     4 , FI, VOBWS, RS, TANS )                                           RIGID   6
C                                                                        RIGID   7
      DIMENSION OMEGA(1), TIME(1), GEOM(1)                               RIGID   8
      COMPLEX CPT(1), CP(NFORCE,1)                                       RIGID   9
      DIMENSION IBOXES(1), ISBES(2,1), Q(1), GEOP(MBOX,1), GEOS(MSBE,1) RIGID  10
      COMPLEX D(MBOX,1), DB(MSBE,1)                                      RIGID  11
      DIMENSION CR(3,1)                                                  RIGID  12
      DIMENSION DP(1), RHO(1), VG(1), PLOTF(1), PLOTA(1)                 RIGID  13
      COMPLEX FTG(1)                                                     RIGID  14
      DIMENSION P(NFORCE,1)                                              RIGID  15
      COMPLEX FI(NFORCE,1)                                               RIGID  16
      DIMENSION VOBWS(1), RS(1), TANS(1)                                 RIGID  17
C                                                                        RIGID  18
      COMMON /ZZZ/HEDR(48),NIN,NOUT,KROW,LINES,IPRNT,NER                 ZZZ     2
C                                                                        RIGID  20
      COMMON /XTF/NF(100)                                                RIGID  21
      EQUIVALENCE (NF(1),AMACH),     (NF(3),VSS)                         RIGID  22
      EQUIVALENCE (NF(6),NSYM),      (NF(9),NFREQ),     (NF(22),IPREQ)   RIGID  23
     1     ,      (NF(12),BZERO),    (NF(13),RHOO),     (NF(14),SIGMA)   RIGID  24
     2     ,      (NF(18),VTFPS)                                         RIGID  25
     3     ,      (NF(41),NK),       (NF(42),NG)                         RIGID  26
     4     ,      (NF(62),TIMEMX),   (NF(97),DELT)                       RIGID  27
     9     ,      (NF(100),SIZFCT)                                       RIGID  28
C                                                                        RIGID  29
      DATA TWOPI/6.2831853/                                              RIGID  30
C                                                                        RIGID  31
    3 FORMAT (1H0,20X,*RIGID TIME RESPONSE PARAMETERS*                   RIGID  32
     1 /1H0,20X,*ORIENTATION NO. =*,I4                                   RIGID  33
     2 /1H ,20X,*MAX. RESP. TIME =*,F10.4                                RIGID  34
     2 /1H ,20X,*NO. TIME RSP.PTS=*,I5                                   RIGID  35
     2 /1H ,20X,*GAMX =*,E12.5                                           RIGID  36
     2 /1H ,20X,*GAMY =*,E12.5                                           RIGID  37
     2 /1H ,20X,*GAMZ =*,E12.5                                           RIGID  38
     2 /1H ,20X,*TAU =*,F10.4,* SECS*                                    RIGID  39
     2 /1H ,20X,*XINT =*,F10.2                                           RIGID  40
     2 /1H ,20X,*YINT =*,F10.2                                           RIGID  41
     2 /1H ,20X,*ZINT =*,F10.2)                                          RIGID  42
   10 FORMAT (6I12)                                                      RIGID  43
   20 FORMAT (6F12.0)                                                    RIGID  44
   30 FORMAT (1H0,3X1HI,9X7HTIME(I),11X5HDP(I),10X6HRHO(I),11X5HVG(I)   RIGID  45
     1 // (1XI4,4E16.6) )                                                RIGID  46
   31 FORMAT (1H0,7X3HSTA,7X3HBOX,17X1HX,9X1HY,9X1HZ,6X4HAREA)          RIGID  47
   32 FORMAT (1X,2I10,10X4F10.2)                                         RIGID  48
   33 FORMAT (1H0,7X3HSTA,6X4HBODY,1X9HDIRECTION,7X1HX,9X1HY,9X1HZ      RIGID  49
     1 , 6X6HLENGTH)                                                     RIGID  50
   34 FORMAT (1X,3I10,4F10.2)                                            RIGID  51
   38 FORMAT (1H0,*INPUT GENERALIZED COORDINATES*                        RIGID  52
     1 // (1X,I4,E16.6))                                                 RIGID  53
```

```
40 FORMAT (1H0,20X,*TIME RESPONSE*)
41 FORMAT (1H0,20X,F8.4,* RESPONSE SECS*,2X,F8.4,* HZ COUNTED*)
42 FORMAT (1H0,2X,*TIME  *,8(7X ,*STA*,I3))
43 FORMAT (1H ,F8.4,8(1X,E12.5))
50 FORMAT (1H0,20X,*FREQUENCY RESPONSE*)
52 FORMAT (1H0,19X,*STA*,I3,3(20X,*STA*,I3)
  1 / 5X,*FREQ*,6X,*REAL*,9X,*IMAG*,3(9X,*REAL*,9X,*IMAG*) )
C
   READ (NIN,10) (IBOXES(I),I=1,NBOX)
   READ (NIN,10) ((ISBES(I,J),I=1,2),J=1,NSBE)
   READ (NIN,20) (Q(I),I=1,NSYM)
   WRITE (NOUT,38) (I,Q(I),I=1,NSYM)
C
   FACTOR = SIZFCT**2
   RHOA = SIGMA*RHOO
   DYNP = SIGMA*RHOO*VTFPS**2*FACTOR/288.
C
   CALL RDAERO (4HGEOM,0,GEOP,NOUT,NER)
   DO 70 I=1,NBOX
   II = IBOXES(I)
70 GEOM(I) = GEOP(II)
C
   CALL RDAERO (4HGEO1,0,GEOP,NOUT,NER)
   DO 75 I=1,MBOX
   GEOP(I,4) = GEOP(I,5) - GEOP(I,2)
75 GEOP(I,5) = GEOP(I,6) - GEOP(I,3)
C
   CALL RDAERO (4HGEO3,0,GEOP,NOUT,NER)
C
   CALL RDAERO (4HGEOS,0,GEOS,NOUT,NER)
   I = NBOX
   DO 80 N=1,NSBE
   I = I+1
   II = ISBES(1,N)
80 GFUM(I) = GEOS(II,4) - GEOS(II,1)
C
   CALL HEADNG
   WRITE (NOUT,31)
   DO 82 I=1,NBOX
   N = IBOXES(I)
   X = GEOP(N,1)
   Y = GEOP(N,2)
   Z = GEOP(N,3)
82 WRITE (NOUT,32)  I,IBOXES(I),X,Y,Z,GEOM(I)
   N = NBOX
   WRITE (NOUT,33)
   DO 55 J=1,NSBE
   N = N+1
   I = ISBES(1,J)
   X = (GEOS(I,1)+GEOS(I,4))/2.0
   Y = (GEOS(I,2)+GEOS(I,5))/2.0
   Z = (GEOS(I,3)+GEOS(I,6))/2.0
```

```
   55 WRITE (NOUT,34) N, (ISBES(I,J),I=1,2),X,Y,Z,GEOM(N)
C
      CALL RDAERO (4HCR   ,0,CR,NOUT,NER)
      GAMX = CR(1,NOR)
      GAMY = CR(2,NOR)
      GAMZ = CR(3,NOR)
C
      CALL BLINT (MBOX,MSBE
     1 , GEOP,GEOS,GAMX,GAMY,GAMZ
     2 , ALI,X,Y,Z )
C
C     NOW THE TIME OF BURST INTERCEPT (WRT ORGIN OF AAS) IS
      TAU = (ALI*SIZFCT/12.)/(VSS*(1.0+GAMX*AMACH))
      IF (TAU.GT.0) TAU=0.
C
      WRITE (NOUT,3) NOR,TIMEMX,NTHRSP,GAMX,GAMY,GAMZ,TAU,X,Y,Z
C
      CALL RDAERO (4HRK   ,0,VOBWS,NOUT,NER)
      DO 90 I=1,NK
      IF (VOBWS(I).EQ.0.0) VOBWS(I)=1.E-10
   90 VOBWS(I) = 1./VOBWS(I)
C
      DO 60 K=1,NFREQ
      OMEGA(K) = OMEGA(K)*TWOPI
      IF (OMEGA(K).LE.0.0) OMEGA(K) = 1.E-6
   60 CONTINUE
      OMEGMX = OMEGA(NFREQ)
      FRQMX = OMEGMX/TWOPI
C
      CALL RDAERO (4HDPSP,1,D,NOUT,NER)
C
      DO 105 I=1,NBOX
      II = IBOXES(I)
      CPT(I) = 0.0
      DO 105 J=1,NSYM
  105 CPT(I) = CPT(I) + DYNP*D(II,J)*Q(J)
C
      CALL RDAERO (4HDZSP,1,DB,NOUT,NER)
C
      DO 120 N=1,NSBE
      IDIR = ISBES(2,N)
      IF (IDIR.EQ.2) GO TO 120
      II = ISBES(1,N)
      I = N + NBOX
      CPT(I) = 0.0
      DO 110 J=1,NSYM
  110 CPT(I) = CPT(I) + DYNP*DB(II,J)*Q(J)
  120 CONTINUE
C
      CALL RDAERO (4HDYSP,1,DB,NOUT,NER)
C
      DO 150 N=1,NSBE
```

```
      IDIR = ISBES(2,N)
      IF (IDIR.EQ.1) GO TO 150
      II = ISBES(1,N)
      I = N + NBOX
      CPT(I) = 0.0
      DO 140 J=1,NSYM
  140 CPT(I) = CPT(I) + DYNP*DB(II,J)*Q(J)
  150 CONTINUE
C
      CALL HEADNG
      CALL PRNT (CPT,NFORCE,1,NFORCE,2,12HTRIM FORCES ,3)
C
      DO 250 K=1,NK
C
      CALL RDAERO (4HFPSP,K,D,NOUT,NER)
C
      DO 220 I=1,NBOX
      II = IBOXES(I)
  220 FI(I,K) = D(II,NOR)
C
      CALL RDAERO (4HFZSP,K,DB,NOUT,NER)
C
      DO 230 N=1,NSBE
      IDIR = ISBES(2,N)
      IF (IDIR.EQ.2) GO TO 230
      II = ISBES(1,N)
      I = N+NBOX
      FI(I,K) = DB(II,NOR)
  230 CONTINUE
C
      CALL RDAERO (4HFYSP,K,DB,NOUT,NER)
C
      DO 240 N=1,NSBE
      IDIR = ISBES(2,N)
      IF (IDIR.EQ.1) GO TO 240
      II = ISBES(1,N)
      I = N+NBOX
      FI(I,K) = DB(II,NOR)
  240 CONTINUE
C
  250 CONTINUE
C
      CALL        GSINT (NFREQ,NFORCE,NK,IPREQ
     1 , OMEGA,CP,FI,VOBWS,RS,TANS)
C
      LINES = KROW+1
      DO 253 JS=1,NFORCE,4
      JF = MINO (JS+3,NFORCE)
      IF (LINES+4.LE.KROW) GO TO 252
      CALL HEADNG
      WRITE (NOUT,50)
      LINES = LINES+2
```

```
  252 WRITE (NOUT,52) (J,J=JS,JF)
      LINES = LINES+3
      DO 253 K=1,NFREQ
      FREQ = OMEGA(K)/TWOPI
      LINES = LINES+1
      IF (LINES.LE.KROW) GO TO 253
      CALL HEADNG
      WRITE (NOUT,52) (J,J=JS,JF)
      LINES = LINES+4
  253 WRITE (NOUT,43) FREQ, (CP(J,K),J=JS,JF)
C
      CALL HEADNG
      DO 254 I=1,NTMGST
  254 READ (NIN,20) TIME(I), DP(I), RHO(I), VG(I)
      WRITE (NOUT,30) (I,TIME(I),DP(I),RHO(I),VG(I),I=1,NTMGST)
C
      CALL PLT1 ( NTMGST,TIME,VG,1)
      WRITE (NOUT,255)
  255 FORMAT (1H0,20X,*INPUT GUST TIME HISTORY*)
C
      DO 256 I=1,NTMGST
  256 TIME(I) = TIME(I) - TAU
C
      CALL TKFFT (FTG,NTMGST,OMEGA,VG,RHO,RHOA,NFREQ
     1 , PLOTF,PLOTA,TIME,GZERO,ALPHA,BETA)
C
      IF (IPLOT.EQ.0) GO TO 290
C
      DO 260 J=1,NFREQ
      PLOTF(J) = OMEGA(J)/TWOPI
  260 PLOTA(J) = SQRT(REAL(FTG(J))**2+AIMAG(FTG(J))**2)
C
      CALL PLT1(NFREQ,PLOTF,PLOTA,1)
      WRITE (NOUT,261)
  261 FORMAT (1H0,20X,21HTRANSFORM OF RHO*GUST )
C
  290 DO 300 J=1,NFREQ
      DO 300 I=1,NFORCE
  300 CP(I,J) = CP(I,J)*FTG(J)*DYNP/VTFPS
C
C     NOW GET SOLN TIMES
      IND=0
      DELTT=DELT
      TI=-DELT
      DO 100 I=1,NTMRSP
      TI=TI+DELTT
      TIME(I)=TI
      IF(IND.GT.1)GO TO 100
      IF(TI.LT.0.25)GO TO 100
      IF(TI.LE.1.0)GO TO 99
      IND=2
      TI=TI-DELTT
```

```
      DELTT=DELTT*2.0
      TI=TI+DELTT
      TIME(I)=TI
      GO TO 100
   99 IF(IND.EQ.1)GC TO 100
      IND=1
      TI=TI-DELTT
      DELTT=DELTT*5.0
      TI=TI+DELTT
      TIME(I)=TI
  100 CONTINUE
C
      IF(TIME(NTMRSP).GT.TIMEMX)TIME(NTMRSP)=TIMEMX
C
      CALL IFT (NFREQ,OMEGA,OMEGMX,0.0,0,NFORCE,CP,NTMRSP,TIME,P,TIMEMX)
C
      DO 310 I=1,NFORCE
      DO 310 J=1,NTMRSP
  310 P(I,J) = (P(I,J) + CPT(I))/GFOM(I)
C
      LINES = KROW+1
      DO 820 JS=1,NFORCE,8
      JF = MINO(JS+7,NFORCE)
      IF (LINES+3.LE.KROW) GO TO 810
      CALL HEADNG
      WRITE (NOUT,40)
      WRITE (NOUT,41) TIMEMX,FRQMX
      LINES = LINES+4
  810 WRITE (NOUT,42) (J,J=JS,JF)
      LINES = LINES+2
      DO 820 K=1,NTMRSP
      LINES = LINES+1
      IF (LINES.LE.KROW) GO TO 82C
      CALL HEADNG
      WRITE (NOUT,42) (J,J=JS,JF)
      LINES = LINES+3
  820 WRITE (NOUT,43) TIME(K), (P(J,K),J=JS,JF)
C
      IF (IPLOT.EQ.0) GO TO 90C
      DO 850 I=1,NFORCE
      DO 840 J=1,NTMRSP
  840 PLOTA(J) = P(I,J)
      CALL PLT1 (NTMRSP,TIME,PLOTA,1)
      WRITE (NOUT,830) I
  830 FORMAT (1H0,20X,*TIME RESPONSE FOR STATION*,I4)
  850 CONTINUE
C
  900 RETURN
      END
```

```
      SUBROUTINE CMERGE                                                 CMERGE 2
C                                                                       CMERGE 3
      COMMON NAA,A(1)                                                   CMERGE 4
      COMMON /ZZZ/HEDR(48),NIN,NOUT,KROW,LINES,IPRNT,NER                ZZZ    2
C                                                                       CMERGE 6
      DIMENSION IHD(50)                                                 CMERGE 7
C                                                                       CMERGE 8
   10 FORMAT (6I12)                                                     CMERGE 9
   20 FORMAT (6F12.0)                                                   CMERGE10
   30 FORMAT (///10X,,*LIST OF REDUCED FREQUENCIES TO BE MERGED FROM TAPCMERGE11
     1E FT*,I2//(5X,6F15.6))                                            CMERGE12
C                                                                       CMERGE13
      NTAPE1 = 17                                                       CMERGE14
      NTAPE2 = 18                                                       CMERGE15
      NTAPE3 = 19                                                       CMERGE16
      REWIND NTAPE1                                                     CMERGE17
      REWIND NTAPE2                                                     CMERGE18
      REWIND NTAPE3                                                     CMERGE19
C                                                                       CMERGE20
      READ (NIN,10) INK1, INK2                                          CMERGE21
C                                                                       CMERGE22
      READ (NTAPE1) IHD                                                 CMERGE23
      NK1 = IHD(2)                                                      CMERGE24
      READ (NTAPE2) IHD                                                 CMERGE25
      NK2 = IHD(2)                                                      CMERGE26
C                                                                       CMERGE27
      L1 = NAA                                                          CMERGE28
      L2 = L1 + INK1                                                    CMERGE29
      L3 = L2 + INK2                                                    CMERGE30
      L4 = L3 + INK1 + INK2                                             CMERGE31
      L5 = L4 + NK1                                                     CMERGE32
      L6 = L5 + NK2                                                     CMERGE33
C                                                                       CMERGE34
      CALL HEADNG                                                       CMERGE35
      LAST = L2-1                                                       CMERGE36
      READ (NIN,20)              (A(I),I=L1,LAST)                       CMERGE37
      WRITE (NOUT,30) NTAPE1,(A(I),I=L1,LAST)                           CMERGE38
      LAST = L3-1                                                       CMERGE39
      READ (NIN,20)             (A(I),I=L2,LAST)                        CMERGE40
      WRITE (NOUT,30) NTAPE2,(A(I),I=L2,LAST)                           CMERGE41
C                                                                       CMERGE42
      CALL MERGE (NTAPE1,NTAPE2,NTAPE3,NOUT,INK1,INK2                   CMERGE43
     1 ,  A(L1),A(L2),A(L3),A(L4),A(L5),A(L6)  )                        CMERGE44
C                                                                       CMERGE45
      RETURN                                                            CMERGE46
      END                                                               CMERGE47
```

```
      SUBROUTINE COPY( A , L , NI , NO )
      DIMENSION  A ( L )
C
      READ   (NI )      A
      WRITE  (NO )      A
C
      RETURN
      END
```

```
      SUBROUTINE FCOPY (NTAPE1, NTAPE2, NTAPE3, NK1   , NK2   , NK    , FCOPY  2
     1                  RK1   , RK2   , RK    , NSYM  , NASYM , NGUST , NBOX  ; FCOPY  3
     2                  NSBE  ,         NTP6  ,                 TEMP          ) FCOPY  4
      DIMENSION  RK1(1), RK2(1), RK(1),                 TEMP(1)                FCOPY  5
C                                                                               FCOPY  6
      NSET       =  3                                                           FCOPY  7
      IF  (NSBE .EQ. 0)  NSET = 1                                               FCOPY  8
      NRECS      =  NSET * (NSYM + NGUST + 1)                                   FCOPY  9
      NRECA      =  NSET * (NASYM+ NGUST + 1)                                   FCOPY 10
      IF  (NASYM .EQ. 0)  NRECA = 0                                             FCOPY 11
      ISKIP      =  NRECS + NRECA                                               FCOPY 12
      NSKIP = 6                                                                 FCOPY 13
      IF (NSBE.NE.0) NSKIP = 7                                                  FCOPY 14
      NSKIP      =  NSKIP + NSET * (NSYM + NASYM)                               FCOPY 15
      ID01       =  0                                                           FCOPY 16
      ID02       =  0                                                           FCOPY 17
      INCR       =  0                                                           FCOPY 18
C                                                                               FCOPY 19
C        MERGE THE FORCE-  AND GENERALIZED FORCE MATRICES FROM THE              FCOPY 20
C        TWO INPUT TAPES  (SKIPPING OVER ONES NOT TO BE SAVED)                  FCOPY 21
C        ONTO THE OUTPUT TAPE  NTAPE3                                           FCOPY 22
C                                                                               FCOPY 23
      K1         =  1                                                           FCOPY 24
      K2         =  1                                                           FCOPY 25
      IF  (NK1 .EQ. 1)  GO TO  20                                               FCOPY 26
      IF  (RK1(1) .GT. RK1(2))  INCR = 1                                        FCOPY 27
   20 IF  (NK2 .EQ. 1)  GO TO  30                                               FCOPY 28
      IF  (RK2(1) .GT. RK2(2))  INCR = 1                                        FCOPY 29
   30 CONTINUE                                                                  FCOPY 30
C                                                                               FCOPY 31
C     START OF FREQUENCY LOOP                                                   FCOPY 32
C                                                                               FCOPY 33
      DO  210  JK = 1, NK                                                       FCOPY 34
      FREQ       =  RK(JK)                                                      FCOPY 35
   40 CONTINUE                                                                  FCOPY 36
      IGO        =  1                                                           FCOPY 37
      NTAPE      =  NTAPE1                                                      FCOPY 38
      RKRUN      = RK1(K1)                                                      FCOPY 39
   50 CONTINUE                                                                  FCOPY 40
C                                                                               FCOPY 41
      IF  (FREQ .EQ. RKRUN)  GO TO  110                                         FCOPY 42
      GO TO  (60,70),  IGO                                                      FCOPY 43
   60 CONTINUE                                                                  FCOPY 44
      IF  (K1 .EQ. NK1)  GO TO  100                                             FCOPY 45
      IF  (FREQ .LT. RKRUN)  GO TO  100                                         FCOPY 46
      K1         =  K1 + 1                                                      FCOPY 47
      RKRUN      =  RK1(K1)                                                     FCOPY 48
      GO TO  80                                                                 FCOPY 49
   70 IF  (K2 .EQ. NK2 .OR. FREQ. LT. RKRUN)  GO TO  220                        FCOPY 50
C                                                                               FCOPY 51
      K2         =  K2 + 1                                                      FCOPY 52
      RKRUN      =  RK2(K2)                                                     FCOPY 53
```

```
   80 CONTINUE
C
C         SKIP OVER A SET OF MATRICES THAT IS NOT TO BE SAVED
C
      DO  90  ISK = 1, ISKIP
      READ   (NTAPE)
   90 CONTINUE
C
      GO TO  (40,50),   IGO
  100 CONTINUE
      IGO        =   2
      NTAPE      =   NTAPE2
      RKRUN      =   RK2(K2)
      GO TO   50
C
  110 CONTINUE
C
C         COPY THIS SET OF FORCE-  AND GENERALIZED FORCE MATRICES
C
      NREC       =   NSYM + NGUST
      DO   140   I = 1, 2
      LENGTH     =   NBOX * 2
      DO   130   NST = 1, NSET
      DO   120   NRC = 1, NREC
C
      CALL         COPY( TEMP, LENGTH, NTAPE, NTAPE3 )
C
  120 CONTINUE
      IF   (NSBE .NE. 0)  LENGTH = NSBE * 2
  130 CONTINUE
C
      IF   (NASYM .EQ. 0)   GO TO   150
      NREC       =   NASYM + NGUST
  140 CONTINUE
C
  150 CONTINUE
      NDIM       =   NSYM * (NSYM + NGUST) * 2
      DO   170   I = 1, 2
      DO   160   N = 1, NSET
C
      CALL         COPY( TEMP, NDIM, NTAPE, NTAPE3 )
C
  160 CONTINUE
      IF   (NASYM .EQ. 0)   GO TO   180
      NDIM       =   NASYM* (NASYM+ NGUST) * 2
  170 CONTINUE
C
  180 CONTINUE
C
      IF   (JK .EQ. NK)   GO TO   210
      IF   (INCR .EQ. 1)   GO TO   220
      GO TO   (190,200),   IGO
```

```
  190 IF   (K1 .EQ. NK1)  GO TO  220                                          FCOPY106
      K1         =  K1 + 1                                                    FCOPY107
      GO TO  210                                                              FCOPY108
C                                                                             FCOPY109
  200 IF   (K2 .EQ. NK2)  GO TO  220                                          FCOPY110
      K2         =  K2 + 1                                                    FCOPY111
  205 FORMAT (1H ,*FORCES WRITTEN TO OUTPUT FOR K=*,F10.6)                    FCOPY112
  210 WRITE (NTP6,205) FREQ                                                   FCOPY113
C                                                                             FCOPY114
C     END   OF FREQUENCY LOOP                                                 FCOPY115
C                                                                             FCOPY116
      RETURN                                                                  FCOPY117
C                                                                             FCOPY118
  220 CONTINUE                                                                FCOPY119
C                                                                             FCOPY120
C          REWIND  NTAPE  AND POSITION IT FOR THE READING OF THE FIRST        FCOPY121
C          SET OF FORCE-  AND GENERALIZED FORCE MATRICES                      FCOPY122
C                                                                             FCOPY123
      REWIND  NTAPE                                                           FCOPY124
      DO  230  NSK = 1, NSKIP                                                 FCOPY125
      READ   (NTAPE)                                                          FCOPY126
  230 CONTINUE                                                                FCOPY127
C                                                                             FCOPY128
      GO TO  (240,250),   IGO                                                 FCOPY129
  240 CONTINUE                                                                FCOPY130
      IDO1       =  IDO1 + 1                                                  FCOPY131
      K1         =  1                                                         FCOPY132
      IF   (FREQ .EQ. RKRUN)  GO TO  210                                      FCOPY133
      GO TO  40                                                               FCOPY134
  250 CONTINUE                                                                FCOPY135
      IDO2       =  IDO2 + 1                                                  FCOPY136
      K2         =  1                                                         FCOPY137
      IF   (FREQ .EQ. RKRUN)  GO TO  210                                      FCOPY138
      GO TO  100                                                              FCOPY139
C                                                                             FCOPY140
      END                                                                     FCOPY141
```

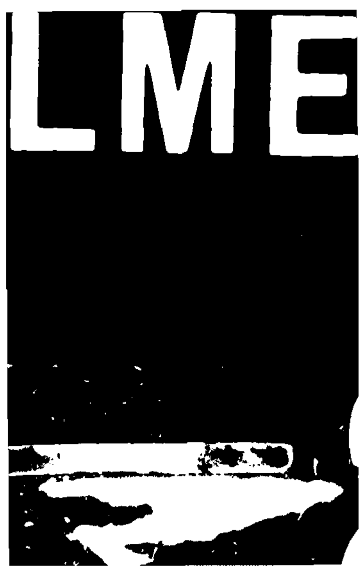
```
      SUBROUTINE MERGE(NTAPE1, NTAPE2, NTAPE3, NTP6, INK1, INK2,
     1     RKIN1, RKIN2,  RK   , RK1    , RK2    ,                TEMP    )
C
      DIMENSION  IDH1(50), IDH2(50), RKIN1(1), RKIN2(1),
     1           RK1(1)   , RK2(1)   , RK(1)    , TEMP(1) ,     NDIM(10)
C
   10 FORMAT   ( ///10X, 32H*** INCOMPATIBLE INPUT TAPES ***    / )
   20 FORMAT   ( /10X, 9HNSYM1     =, I4, 11H   NSYM2     =, I4, 7H -STOP-   )
   30 FORMAT   ( /10X, 9HNASYM1    =, I4, 11H   NASYM2    =, I4, 7H -STOP-   )
   40 FORMAT   ( /10X, 9HNGUST1    =, I4, 11H   NGUST2    =, I4, 7H -STOP-   )
   50 FORMAT   ( /10X, 9HNB1       =, I4, 11H   NB2       =, I4, 7H -STOP-   )
   60 FORMAT   ( /10X, 9HNBOX1     =, I4, 11H   NBOX2     =, I4, 7H -STOP-   )
   70 FORMAT   ( /10X, 9HNSBE1     =, I4, 11H   NSBE2     =, I4, 7H -STOP-   )
   80 FORMAT   ( /10X, 9HNSTRIP1 =, I4, 11H   NSTRIP2 =, I4, 7H -STOP-   )
   90 FORMAT   ( /10X, 9HMAXSTR1 =, I4, 11H   MAXSTR2 =, I4, 7H -STOP-   )
  100 FORMAT   ( /10X, 9HNP1       =, I4, 11H   NP2       =, I4, 7H -STOP-   )
  110 FORMAT   ( ///10X, 11H---- FREQ =, F10.6, 24H WAS NOT FOUND ON TAPE
     1         , I2 // 10X, 20HCASE IS TERMINATED                  )
C
      NK        =   INK1 + INK2
      REWIND   NTAPE1
      REWIND   NTAPE2
      REWIND   NTAPE3
C
      READ     (NTAPE1)   IDH1
      READ     (NTAPE2)   IDH2
C
      DO   120   I = 3, 8
      IF   (IDH1(I) .EQ. IDH2(I))   GO TO   120
      IX        =   I - 2
      GO TO   130
C
  120 CONTINUE
      GO TO   200
C
  130 CONTINUE
      WRITE    (NTP6,10)
C
      GO TO    (140,150,160,170,180,190),   IX
  140 WRITE    (NTP6,20)   IDH1(3),   IDH2(3)
      STOP
  150 WRITE    (NTP6,30)   IDH1(4),   IDH2(4)
      STOP
  160 WRITE    (NTP6,40)   IDH1(5),   IDH2(5)
      STOP
  170 WRITE    (NTP6,50)   IDH1(6),   IDH2(6)
      STOP
  180 WRITE    (NTP6,60)   IDH1(7),   IDH2(7)
      STOP
  190 WRITE    (NTP6,70)   IDH1(8),   IDH2(8)
      STOP
C
```

```
  200 CONTINUE
C
      NK1       =   IDH1(2)
      NK2       =   IDH2(2)
      IDH1(2) = NK
      CALL WRAERO (4HHEAD,0,IDH1,NTP6,NER)
C
      NSYM      =   IDH1(3)
      NASYM     =   IDH1(4)
      NGUST     =   IDH1(5)
      NB        =   IDH1(6)
      NBOX      =   IDH1(7)
      NSBE      =   IDH1(8)
      NSTRIP    =   IDH1(9)
      MAXSTR    =   IDH1(10)
      NP        =   IDH1(11)
C
      NGEOM     =   3*NBOX + 8*NSTRIP + MAXSTR + 2*NP + 3*NB
      NDIM(1)   =   NGEOM
      NDIM(2)   =   6 * NBOX
      NDIM(3)   =   3 * NBOX
      NDIM(4)   =   6 * NSBE
      NDIM(5)   =   3 * NGUST
C
C         COPY RECORD TYPES  2  THROUGH  6  FROM NTAPE1  ONTO  NTAPE3
C
      DO  210  J = 1, 5
      LENGTH    =  NDIM(J)
      IF  (LENGTH .EQ. 0)  GO TO  210
C
      READ   (NTAPE2)
      CALL        COPY( TEMP, LENGTH, NTAPE1, NTAPE3 )
C
  210 CONTINUE
C
C         READ FREQUENCY ARRAYS FROM THE TWO INPUT TAPES
C
      READ   (NTAPE1)  (RK1(I),  I = 1, NK1)
      READ   (NTAPE2)  (RK2(I),  I = 1, NK2)
C
C         CHECK CARD-INPUT FREQUENCY ARRAYS AGAINST FREQUENCIES THAT
C         ARE AVAILABLE ON THE TWO INPUT TAPES
C
      NTAPE     =  NTAPE1
      INK       =  INK1
      NKR       =  NK1
      DO  310  IGO = 1, 2
      NKMAX = 0
      DO  300  K = 1, INK
      GO TO  (220,230),  IGO
  220 FREQ      =  RKIN1(K)
      GO TO  240
```

```
  230 FREQ      =  RKIN2(K)
C
  240 CONTINUE
C
      DO  280  J = 1, NKR
      JMAX = J
      GO TO  (250,260),  IGO
  250 RPICK     =  RK1(J)
      GO TO  270
  260 RPICK     =  RK2(J)
C
  270 CONTINUE
       IF  (FREQ .EQ. RPICK)  GO TO  290
C
  280 CONTINUE
C
C         THIS (CARD-INPUT)  FREQUENCY IS NOT AVAILABLE ON TAPE -- STOP
C
      WRITE  (NTP6,110)  FREQ, NTAPE
C
      STOP
C
  290 IF (JMAX.GT.NKMAX) NKMAX = JMAX
  300 CONTINUE
      IF  (IGO.EQ.1) NK1 = NKMAX
      IF  (IGO.EQ.2) NK2 = NKMAX
      NTAPE     =  NTAPE2
      INK       =  INK2
      NKR       =  NK2
C
  310 CONTINUE
C
C         MOVE THE TWO INPUT-FREQUENCY-ARRAYS INTO  TEMP,  THEN SORT
C          IN ASCENDING ORDER  AND SAVE IN ARRAY  RK
C
      DO  330  K = 1, NK
      IF  (K .GT. INK1)  GO TO  320
      TEMP(K)   =  RKIN1(K)
      GO TO  330
C
  320 CONTINUE
      I         =  K - INK1
      TEMP(K)   =  RKIN2(I)
  330 CONTINUE
      J         =  NK
  340 CONTINUE
      RKMAX     =  0.0
      DO  350  K = 1, NK
      IF  (RKMAX .LT. TEMP(K))  RKMAX =  TEMP(K)
  350 CONTINUE
C
      RK(J)     =  RKMAX
```

```
      IF   (J .EQ. 1)  GO TO  360                                           MERGE158
      DO   352  K = 1, NK                                                   MERGE159
      L         =  K                                                        MERGE160
      IF   (RKMAX .EQ. TEMP(L))  GO TO  354                                 MERGE161
  352 CONTINUE                                                              MERGE162
      STOP  16                                                              MERGE163
C                                                                           MERGE164
  354 TEMP(L)   =  0.0                                                      MERGE165
C                                                                           MERGE166
      J         =  J - 1                                                    MERGE167
      GO TO  340                                                            MERGE168
C                                                                           MERGE169
  360 CONTINUE                                                              MERGE170
C                                                                           MERGE171
C         WRITE SORTED FREQUENCY ARRAY ON OUTPUT TAPE  NTAPE3              MERGE172
C                                                                           MERGE173
      WRITE  (NTAPE3)  (RK(I),  I = 1, NK)                                  MERGE174
C                                                                           MERGE175
C         COPY THE MATRICES  HPS, HPA, HZS, HZA, HYS, HYA                   MERGE176
C                                                                           MERGE177
      LENGTH    =  NBOX                                                     MERGE178
      DO  400  K = 1, 3                                                     MERGE179
      NREC      =  NSYM                                                     MERGE180
      DO  380  I = 1, 2                                                     MERGE181
      DO  370  N = 1, NREC                                                  MERGE182
C                                                                           MERGE183
      READ   (NTAPE2)                                                       MERGE184
      CALL         COPY( TEMP, LENGTH, NTAPE1, NTAPE3 )                     MERGE185
C                                                                           MERGE186
  370 CONTINUE                                                              MERGE187
C                                                                           MERGE188
      IF   (I .EQ. 2)  GO TO  380                                           MERGE189
      IF   (NASYM .EQ. 0)  GO TO  390                                       MERGE190
      NREC      =  NASYM                                                    MERGE191
  380 CONTINUE                                                              MERGE192
  390 CONTINUE                                                              MERGE193
      IF   (NSBE .EQ. 0)  GO TO  410                                        MERGE194
      LENGTH    =  NSBE                                                     MERGE195
  400 CONTINUE                                                              MERGE196
C                                                                           MERGE197
  410 CONTINUE                                                              MERGE198
      WRITE  (NTP6,420)  NTAPE3,  (RK(I),  I = 1, NK)                       MERGE199
  420 FORMAT  ( //10X,45HHEADER-ITEMS, GEOMETRY, FREQUENCY ARRAY  AND ,    MERGE200
     1         36HTHE H-MATRICES ARE SAVED ON TAPE  FT, I2 /// 10X        , MERGE201
     2         55HBEGIN MERGING THE FORCE- AND GENERALIZED FORCE MATRICESMERGE202
     3         // 10X, 47HTHE COMBINED (SORTED) OUTPUT-FREQUENCY-ARRAY ISMERGE203
     4         // ( 10X, 6F10.6 ) )                                         MERGE204
C                                                                           MERGE205
C         COPY FORCES AND GENERALIZED FORCES    --                          MERGE206
C                                                                           MERGE207
      CALL         FCOPY (NTAPE1, NTAPE2, NTAPE3, NK1   , NK2   , NK    ,  MERGE208
     1             RK1    , RK2   , RK    , NSYM   , NASYM , NGUST , NBOX   , MERGE209
```

```
     2           NSBE  ,     NTP6  ,                 TEMP        ) MERGE210
                                                                   MERGE211
C                                                                  MERGF212
C                                                                  MFRGF213
       RETURN                                                      MERGE214
       END
```


LME